NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

A SYSTEMS ENGINEERING SURVEY OF ARTIFICIAL INTELLIGENCE AND SMART SENSOR NETWORKS IN A NETWORK-CENTRIC ENVIRONMENT

by

David C. Schafer

September 2009

Thesis Co-Advisors:	Rachel Goshorn
	Deborah Goshorn
Second Reader:	Eugene Paulo

Approved for public release, distribution is unlimited

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 2009	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE A Systems Engineering Survey of Artificial Intelligence and Smart Sensor Networks in a Network-Centric Environment			**5. FUNDING NUMBERS**
6. AUTHOR(S) David C. Schafer			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

The implementation of network-centric warfare (NCW) and network-centric operations (NCO) is of paramount importance to the DoD. Fundamentally, the key to implementing NCW/NCO is the accurate obtainment and analysis of critical information to the warfighter. Additionally, the proliferation of sensors, in both types and numbers, is making it apparent that there will simply not be enough military personnel to monitor, analyze and synthesize all pertinent data. It is apparent that a "smart sensor network," or a network of sensors with data analyzed by artificial intelligence (AI), is needed to better facilitate the attainment of the full realization of network-centric operations.

This thesis presents a survey of the information required for individuals who will be involved in the design and acquisition of smart sensor networks, with a focus on systems engineering. The foundations of smart sensor networks are in AI, Distributed AI, multiagent systems, sensor basics, and data fusion. In addition to an examination of the previous topics, this thesis examines what must be done to further the preparedness of systems engineers for better understanding and designing of smart sensor networks.

14. SUBJECT TERMS Smart Sensor Networks, Artificial Intelligence, Distributed Artificial Intelligence, Multiagent Systems, Network-centric Warfare, Network-centric Operations, Systems Engineering, Network-centric Systems Engineering, System of Systems			**15. NUMBER OF PAGES** 113
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89) Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

A SYSTEMS ENGINEERING SURVEY OF ARTIFICIAL INTELLIGENCE AND SMART SENSOR NETWORKS IN A NETWORK-CENTRIC ENVIRONMENT

David C. Schafer
Lieutenant, United States Navy
B.S., Marquette University, 2000

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN SYSTEMS ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
September 2009

Author: David C. Schafer

Approved by: Rachel Goshorn
Thesis Co-Advisor

Deborah Goshorn
Thesis Co-Advisor

Eugene Paulo
Second Reader

David H. Olwell
Chairman, Department of Systems Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The implementation of network-centric warfare (NCW) and network-centric operations (NCO) is of paramount importance to the DoD. Fundamentally, the key to implementing NCW/NCO is the accurate obtainment and analysis of critical information to the warfighter. Additionally, the proliferation of sensors, in both types and numbers, is making it apparent that there will simply not be enough military personnel to monitor, analyze and synthesize all pertinent data. It is apparent that a "smart sensor network," or a network of sensors with data analyzed by artificial intelligence (AI), is needed to better facilitate the attainment of the full realization of network-centric operations.

This thesis presents a survey of the information required for individuals who will be involved in the design and acquisition of smart sensor networks, with a focus on systems engineering. The foundations of smart sensor networks are in AI, Distributed AI, multiagent systems, sensor basics, and data fusion. In addition to an examination of the previous topics, this thesis examines what must be done to further the preparedness of systems engineers for better understanding and designing of smart sensor networks.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

The United States military has recently become a more complex entity with a multitude of threats (unknown and known) dispersed across the globe. It is imperative in today's battle spheres of influence to take advantage of both the technology of sensors and the artificial intelligence that automates information extraction from sensors to maintain supremacy over these threats. The Department of Defense (DoD) is confronted with the problem of obtaining data from a plethora of sources; evaluating this data; and then determining the applicability of the data. Network-centric systems provide the foundation upon which the military will be able to gain the upper hand in future conflicts, as network-centric systems are system of systems that work together to collect, fuse and present information in a push/pull network environment. Furthermore, with the increase of sensors and sensor types, there will never be enough people and resources to perform the necessary network-centric operations. Therefore, it is imperative that network-centric systems incorporate artificial intelligence and smart sensor networks for the automation of intelligence in a network-centric system of systems.

In this thesis, the following question is examined: What are the fundamental concepts of artificial intelligence and sensor networks as they relate to network-centric systems? This thesis serves as a stepping-stone into the wide world of distributed artificial intelligence and sensor networks and how they relate to network-centric systems and Network-Centric Warfare and Operations. This thesis will help guide acquirers and systems engineers in making better decisions. Additionally, the information from this thesis will be able to be used in the creation of a class tailored for artificial intelligence and smart sensor networks for network-centric systems.

In this thesis, the two vital elements of sensor networks will be examined: sensor networks (the collection of data) and artificial intelligence (AI) (the evaluation of data). Before delving right into AI and sensor networks, network-centric systems engineering (NCSE), network-centric warfare (NCW) and network-centric operations (NCO) are all discussed. Essentially, smart sensor networks are a Network-Centric Systems Engineering (NCSE) solution to the problem of implementing NCW and NCO. Smart

sensor networks will give the war fighter the tactical advantage of superior situational awareness of all aspects of the battle space. With distributed artificial intelligence (DAI), sensors are able to no longer be seen as standalone entities, but now as subsystems of a system that feeds a system of systems where the result is greater than the sum of the parts.

This thesis also presents a brief history and introduces the basic concepts of AI. Research in AI has gone through many stages since the 1950s, when the term "artificial intelligence" was first coined. In this thesis, the agent concept is used to model AI. An agent is something that gets information from the environment and then acts upon that environment, based on information from the environment. There are various types of agent models that reflect the ways in which analyzes the data received. Additionally, the other backbone of AI, search algorithms, is examined. An understanding of search algorithms is important because the search algorithm used for searching a database will determine the efficiency of an AI application.

DAI and multiagent systems (MAS) are presented as the manner in which to model a smart sensor network. The distributed nature of military forces and sensors calls for the use of DAI to coordinate it all. Additionally, the MAS architecture provides a way of modeling sensor in which sensors act in concert with one another, where all are working towards a common goal (detection and analysis of information). In addition to DAI and MAS, the four basic functions of a smart sensor network's DAI (Detection, Identification, Prediction, and Reaction) are examined.

The recent explosion of microprocessor technology has led to more processing being able to be carried out at the local sensor. The basic architecture of a smart sensor is presented, to include: transducer, signal pre-processing, analog-to-digital conversion, application algorithms, user interface, data storage, and the communication node. While all aspects of the architecture are important, greater emphasis needs to be on the application algorithms, where the AI happens, and the communications interface. The importance of the communications interface is in the sensor protocol used, since it is not important to necessarily have a standard sensor, but it is of the utmost importance to have standard protocols and interfaces for sensors. Various sensor protocols are discussed,

but, the emphasis is on the standardization of a sensor protocol, which the IEEE 1451.2-1997 standard ("IEEE Standard for a Smart Transducer Interface for Sensors and Actuators") attempts to facilitate.

This thesis provides snapshots of various applications and research work conducted on either smart sensor networks and artificial intelligence or both. It is apparent that many other institutions and companies are beginning to see the vast importance of smart sensor networks.

This thesis concludes that the network-centric systems engineer will have a better understanding of how to tackle a design problem involving smart sensor networks armed with the basic knowledge of AI and sensor networks. Based on this conclusion, recommendations for a smart sensor network system design as well as education and research are presented. These recommendations can be used by systems engineers and by those researching smart sensor networks. These recommendations will expedite the realization of a military that can completely and successfully carry out network centric operations.

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

First and foremost, I would like to thank my wife, Shana, and daughter, Eva, for their love, support and encouragement during my time at Naval Postgraduate School. Shana and Eva are the most important people in my life. I love you both.

I would also like to thank Dr. Rachel Goshorn for her guidance and mentoring throughout the research and writing of this thesis. I also thank my co-advisor, Deborah Goshorn, for her invaluable recommendations. Additionally, I thank Dr. Eugene Paulo for his helpful advice.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. OVERVIEW

The United States military has recently become a more complex entity, with a multitude of threats (unknown and known) dispersed across the globe. It is imperative in today's battle spheres of influence to take advantage of both the technology of sensors and the artificial intelligence that automates information extraction from sensors to maintain supremacy over these threats. The domains of the Department of Defense (DoD), including all four services, with emphasis on the Navy, are presented with the daunting problem of obtaining data from a plethora of sources; evaluating this data; and then determining the applicability of the data. Network-centric systems provide the foundation upon which the military will be able to gain the upper hand in future conflicts, as network-centric systems are system of systems that work together to collect, fuse and present information in a push/pull network environment. A network-centric system may view the battle space as a system of systems in the DoD, where each element/system extracts information from a network and serves a unique mission/goal, and shares a common goal of defending the United States. Furthermore, with the increase of sensors and sensor types, there will never be enough facilities, people, bandwidth, and power and weight (for mobile) to perform the necessary network-centric operations. Therefore, it is imperative that network-centric systems incorporate artificial intelligence and smart sensor networks for the automation of intelligence in a network-centric system of systems. The requirement for manual analysis using facilities and humans must be automated and the requirement for bandwidth and power consumption must be compressed.

Thus, this thesis primarily examines the question: What are the fundamental concepts of artificial intelligence and sensor networks as they relate to network-centric systems? This thesis serves as a stepping-stone into the wide world of distributed artificial intelligence and sensor networks, and how they relate to network-centric systems and Network-Centric Warfare and Operations. It helps guide acquirers and systems engineers to make better decisions and makes aware the need to incorporate

artificial intelligence into their knowledge base. Additionally, it is anticipated that elements of this thesis will be able to be incorporated into graduate level study of network-centric systems engineering.

Two essential elements of this issue will be examined: sensor networks (the collection of data) and artificial intelligence (the evaluation of data). When these two subjects are examined through the perspective of systems engineering, the problem becomes much less formidable and easier to handle. In order to obtain a basic understanding of the study of sensor networks and artificial intelligence and why they are essential items in network-centric systems that more than merits further consideration for follow on studies, the below topics will be examined:

- Network-centric systems
- Network-centric warfare and operations
- System of systems engineering
- History of artificial intelligence (AI)
- Basic algorithms and low level classifiers
- Multi-agent systems (MAS)
- Distributed artificial intelligence (DAI)
- Noteworthy research

The intent of this work is to affirm the relevance of sensor networks and artificial intelligence and show the importance of the study/familiarization of these topics is to Navy acquirers. This thesis aptly serves as a primer for the subject matter.

In this section, an overview of the thesis was presented; the next section will discuss what a network-centric system is.

B. WHAT IS A NETWORK-CENTRIC SYSTEM?

A network-centric system is an interconnection of hardware, software and humans that operate over a network (Internet, local area network, intranet, etc.) for the purpose of

accomplishing a specified set of goals. It is important to understand what a network-centric system is, as a smart sensor network (sensor network with AI processing) is considered one.

Network-Centric Systems Engineering (NCSE) is the study of network-centric systems with an emphasis on the systems engineering process. NCSE lays the foundations for a more complete understanding of smart sensor networks. There are four approaches to NCSE, seen in Figure 1 and discussed below:

1. Top-down Approach

The top-down approach provides a broad overview of a system and the services/capabilities it can provide (or the services/capabilities that are wanted). The focus is on the end result (Service Oriented Architecture, Net-Centric Enterprise Services, Communities of Interest, etc.) and the higher level capabilities that are provided by these systems. In this approach, each subsystem is refined in greater detail until the base elements are reached.

2. Bottom-up Approach

The bottom-up approach places much more emphasis on the actual elements of network-centric systems and what services/capabilities they are capable of providing to the war fighter. This approach should be followed when working on a system such as a smart sensor network, and it is the focus of this thesis.

3. Middle Approach

The middle approach is a way of fusing the top-down and bottom up worlds. The middle approach is the realm of smart push (publish) and smart pull (subscribe). Typically, when coming from a top-down approach, the smart pull world is utilized and when coming from the bottom-up approach, the smart push is utilized.

4. Side View Approach

The side view approach is a way of examining how to provide data and communications to (and from) the tactical edge where the user needs or has essential

data, but has a limited means (due to communications, security, by choice, etc.) of access to a network, hence making them a "disadvantaged user."

5. Bringing it all Together

While it is easy to get caught in a stovepipe in focusing on just one of the above approaches, it is essential, especially for those in the NCSE realm, to have an intimate knowledge of all approaches. Without understanding the top-down approach, it is harder to know and understand the desired end state of the user's system; similarly, without understanding the bottom-up approach, it is harder to even know what capabilities are available to the engineer; without understanding the middle approach, the engineer is doomed to develop the system in a vacuum (the top-down and bottom-up worlds will never be connected), and there will be a lack of automation and integration within the system; and finally, without understanding the side-view, the engineer will fail to understand the needs of the end-user. Figure 1 is an abstract representation of the NCSE approaches, where the trunk of the tree is the core principles of network-centric systems engineering, and the limbs represent the four approaches to network-centric systems engineering. The focus of this thesis is to emphasize the bottom-up approach by showing the capabilities and recommendations of AI and sensor networks to network-centric systems. Applying these approaches of NCSE naturally leads to the study of network-centric warfare and operations, which is discussed in the next section.

In this section, the approaches to network-centric systems engineering were presented; the next section will discuss network-centric warfare and operations.

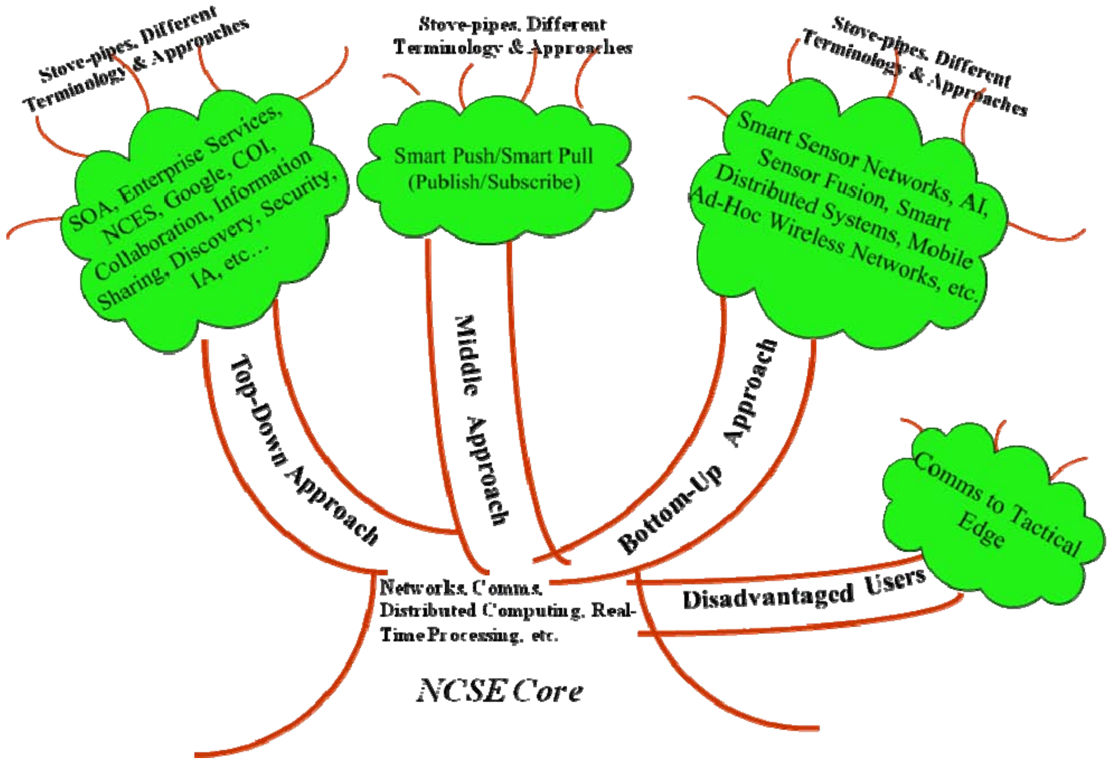

Figure 1. NCSE Approaches. (From [1])

C. WHAT IS NETWORK-CENTRIC WARFARE AND OPERATIONS (NCW/NCO)?

With the evolving nature of warfare in recent years, it has become necessary to define a new battle front and means of gaining leverage in that battle front. Network-centric warfare and operations serve this purpose and are based on the concept of sharing secure, dynamic and effective data between users (war fighters and commanders), based on need and independent of where the users are located (and when they need the information). The below topics will be covered in this section to obtain a better understanding of these developing concepts:

- Network-centric Warfare
- Network-centric Operations

1. Network-centric Warfare

Network-centric warfare (NCW) is considered the military's response to the "Information Age." NCW generally explains the blending of strategies, emerging tactics, techniques, procedures and organizations that a fully or even a partially networked force can employ to create a decisive war fighting advantage [2]. Quality of service in military operations is the impetus behind the concept of NCW. The idea is to take advantage of qualitative data and to automatically assemble a set of capabilities to resolve and drive a particular situation into a desired state. NCW is meant to support the commanders' goals of shared situational awareness, mission effectiveness and as mentioned, quality of service. Quality of service is characterized in [3] as a combination of the information richness, reach and effective value as experienced by the systems users [3].

An obvious requirement for effective NCW is to set up and sustain connectivity between people and equipment engaged in a particular mission, whether or not the participants are geographically co-located. The overall success of the mission is intertwined with the quality of information flow between the elements of the mission (to include the user and equipment) [3].

The four basic tenets of NCW are defined in [2] as:

- "A robustly *networked* force improves information sharing."
- "*Information sharing* enhances the quality of information and shared situational awareness."
- "*Shared situational awareness* enables collaboration and self-synchronization, and enhances sustainability and speed of command.
- "These, in turn, dramatically increase mission effectiveness."

It is clear that the above tenets, when approaching from a NCSE point of view, embody the top-down approach (Figure 1). The implementation and attainment of these tenets could not happen without the bottom-up approach. When looking at the problem from the bottom-up approach, the engineer will be able to identify the tools required to fulfill these tenets. Furthermore, an engineer with knowledge of Artificial Intelligence/

Distributed Artificial Intelligence (AI/DAI) and sensor networks/sensor fusion will be better equipped to accomplishing these tenets.

In addition to understanding the importance of NCW, it is also important to recognize the bathe environments in which NCW takes place. [2] also acknowledges four separate (and overlapping) NCW "Domains of Conflict" (Figure 2):

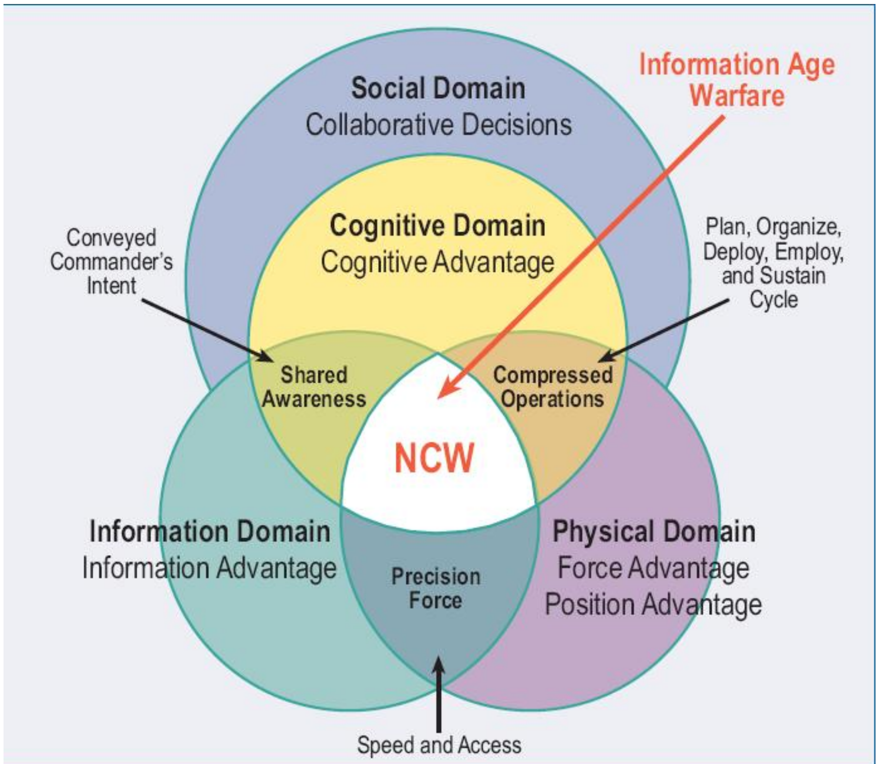

Figure 2. Domains of Conflict. (From [2])

- Physical Domain: this is the traditional paradigm of warfare where a military force (troops, ships, tanks, etc.) is "moved through time and space." This domain is easily measurable, which has led to combat power being traditionally measured in this domain.

- Cognitive Domain: As the name implies, this is considered to be in the "mind of the warfighter." This is the domain in which such ethereal concepts as leadership, morale, tactics, and doctrine emerge.

- Social Domain: This is the domain in which humans interact and exchange the vital information with one another.
- Information Domain: This is the domain in which "information is created, manipulated, and shared." It is the "domain of sensors and the processes for sharing and accessing sensor products as well as "finished" intelligence." This is the domain that this thesis is specifically concerned with. Essentially, it is the domain of smart sensor networks.

2. Network-centric Operations

While NCW is considered the theory of operations, Network-Centric Operations (NCO) is the theory put into action (it is the implementation of NCW). NCO involves the application of the tenets and principles of NCW. A force that implements NCO is "more adaptive, ready to respond to uncertainty in the very dynamic environment of the future at all levels of warfare and across the range of military operations [2]." The operational effectiveness of joint forces increases as forces become more networked, communicate more efficiently, and share common operating picture (shared situational awareness). Vice Admiral (Ret.) Arthur K. Cebrowski made the following remarks to the Defense Writers Group on 23 April 2003: "The things that compel that [a new air-land dynamic] are good *sensors*, networked with good intelligence, disseminated through a robust network of systems which then increases speed [2]." Reference [2] provides some insights into the effectiveness of NCO during Operation Enduring Freedom and Operation Iraqi Freedom.

During Operation Enduring Freedom (OEF) (2001–2002) weapons platforms were successfully networked with sensor platforms. The warfighters attested to the apparent advantages this brought them. Unmanned aerial vehicles (UAVs) began to be used much more during this operation. They were able to provide commanders a near-real time battlefield situational awareness. Many of the lessons that were learned during OEF were applied during in the next major military operation.

Operation Iraqi Freedom (OIF) improved on much of the NCO capabilities from OEF. OIF saw not only the fusion of sensors within a network, but, also a fusion of the

warfighter with networked and joint force. Brigadier General Dennis Moran said: "The ability to move intelligence rapidly from the sensor to either an analytical decision maker or directly to the shooter was the best that we have ever seen… We validated the concept of network-centric warfare [2]."

To conclude, both OEF and OIF have proven that improving the quality of networking results in better information sharing, improved cooperation, and increased speed of command. OEF and OIF will be looked back upon as the dawn of the age of network-centric warfare and operations.

In this section, network-centric warfare and operations were discussed; the next section will address the need for sensor networks and distributed artificial intelligence.

D. WHY ARE SENSOR NETWORKS AND DISTRIBUTED ARTIFICIAL INTELLIGENCE NEEDED?

The obvious answer to the above question is that sensor networks and artificial intelligence are needed to fulfill the goals of NCW. Only through the implementation of smart sensor networks can the NCW tenets be fully implemented. A smart sensor network will give the commander the tactical advantage of having superior situational awareness of all aspects of the battle environment. Because the emphasis is no longer on "who" an adversary is, but rather "how" the adversary might fight, thus broadening the strategic perspective. Smart sensor networks give the commander the enhanced abilities of adaptability and innovative.

Distributed artificial intelligence enables sensors to not be seen just as stand-alone systems, but as a system of systems linked through a common interface and distributed artificial intelligence. The use of DAI links various platforms together (ship, submarine, aircraft, etc.). The concept of "Detection, Interpretation, Prediction, Reaction" (DIPR) (Figure 3) (discussed at more length later) represent the core levels or functions artificial intelligence in a smart sensor network.

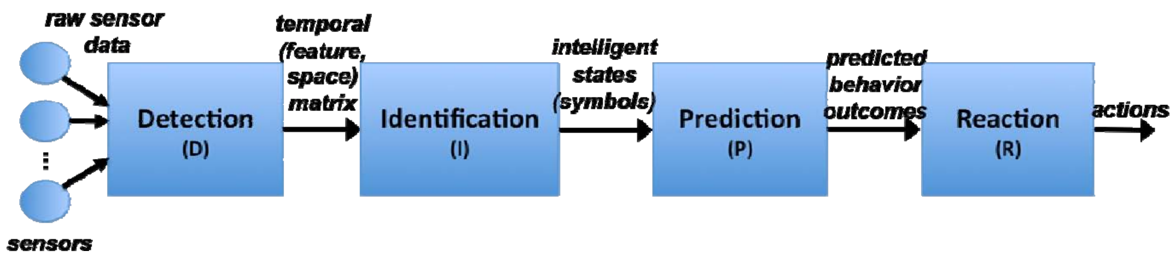

Figure 3. DIPR Process. (From [4])

These functions are, in essence, the requirements of NCO. DIPR mirrors the nature in which humans analyze and react to their environment. This process now brings AI to the sensor network, enabling a smart sensor network, and thus increases the overall effectives of the warfighter. Furthermore, just like agents are a way to model AI, MAS and the DIPR architecture are used to model DAI. The power of MAS, DAI and DIPR is the ability to effectively model a smart sensor network.

In this section, the need for sensor networks and distributed artificial intelligence was discussed; the next section will present the remaining outline of the thesis.

E. THESIS OUTLINE

This section presents succinct overviews of each chapter in this thesis. Each topic builds upon the other and is examined in a logical manner.

1. Artificial Intelligence

A brief history and background on AI will be presented to give the reader an appreciation of the rich history of AI as well as the ups and downs it has gone through. The reader will be introduced to the concepts of agents and some of the various basic algorithms/search paradigms that have been developed for AI. Additionally, there will be a brief review of low- and high-level classifiers.

2. Distributed Artificial Intelligence, DIPR and Multiagent Systems

This chapter will discuss further the importance of DAI as well as delve deeper into the DIPR architecture. Additionally, the reader will be introduced to the elements of a multiagent system, a key infrastructure for DAI.

3. Sensor Networks

This chapter gives an overview of the basic sensor architecture and provides systems engineering perspective on the importance of sensors. Various sensor protocols are discussed, with focus given to the need for standardization. Different types of data fusion are discussed as well as examples of advanced sensor algorithms.

4. Applications

This chapter provides a brief overview of various applications of sensor networks and artificial intelligence. Examples from academic papers to commercial applications are presented.

5. Recommendations

This chapter makes recommendations for how to approach the design and modeling of a smart sensor network. Additionally, recommendations on how to implement a NCSE course dedicated to the study of AI and sensor networks are given.

6. Conclusion and Summary

This chapter concludes the thesis and presents a summary of each chapter.

In this chapter, an overview of the thesis was presented, along with a discussion of network-centric systems engineering approaches, network-centric warfare and operations and the need for smart sensor networks; the next chapter will discuss the foundations of artificial intelligence.

THIS PAGE INTENTIONALLY LEFT BLANK

II. ARTIFICIAL INTELLIGENCE

A. OVERVIEW

Artificial Intelligence (AI) automates the resources to characterize and represent concepts and provides a related reasoning mechanism for use in decision making [5]. As the battle-environment becomes more network-centric, there is a need for more efficient processing of the multitude of sensor data (required by both the end user/war fighter and the commander). It is simply physically impossible for human users alone to filter, analyze, synthesize and communicate to other users the assessment of high volume data. This is where AI can make data collection and analysis much more efficient and reliable. The network-centric systems engineer should have an appreciation and basic understanding of AI concepts. This section will cover the following AI topics:

- A brief history and background
- The basic concepts

In this section, an overview of the chapter was discussed; the next section provides the required background to better understand how AI came into being.

B. A BRIEF HISTORY AND BACKGROUND

The study of AI has gone through many phases since the mid 1950s, with periods of exuberant research and apathetic indifference. Since the late 1980s, the study of AI has become much more prevalent as computing capabilities become ever more powerful.

At this point, it is worth defining what is considered "intelligence" in the scope of AI. Intelligence can be thought of as characteristics of the human mind, to include problem solving, planning, and reasoning. Essentially, intelligence is what gives humans the ability to make the "right decision" (output, with alternatives) given a set of stimuli from their environment (inputs). The key element in differentiating between the intelligence that may be observed in some animals and the intelligence known to exist in humans is that humans have the ability to communicate through a standard language (varied only by regional dialects). So, when the qualifier "artificial" is placed in front of

intelligence, it is meant to describe a computer system that has the capability of performing various functions that might normally be performed by a human (imitates human behavior, Figure 4) [6].

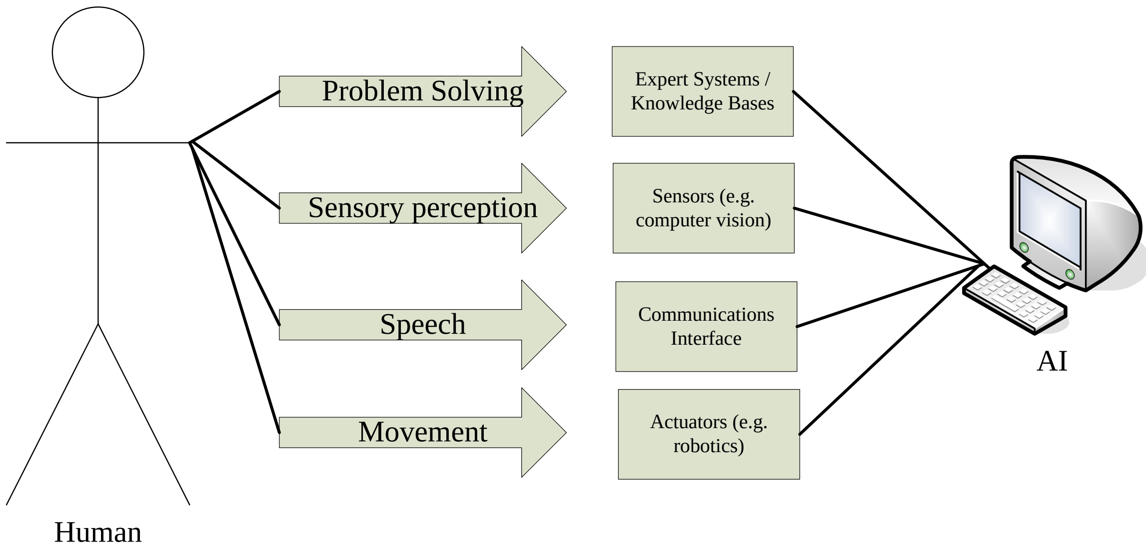

Figure 4. Human / AI Function Parallels. (After [6])

1. Conception (Early 1950s)

The term "artificial intelligence" had yet to be coined, but, an innovative individual, Alan Turing, was one of the first (1950) who asked whether a machine had the ability to think. Turing developed the concept of the abstract machine called the Turing Machine that could solve any basic math problem. Expanding from the idea of the Turing Machine, he went on to theorize that if a computer's response was the same response that would be expected from a human, then the computer could be considered a thinking machine. This test became known as the "Turing Test." If a computer was able to "trick" a human into thinking that it was also a human (through its responses communicated through a computer terminal), then it passed the intelligence test [7].

Other ground breaking work in the early 1950s was performed by Warren McCulloch and Walter Pitts (who are recognized as contributing the first work done for the study of AI). They drew upon the study of the work on the function of neurons in the brain, propositional logic, and Alan Turing's theories on computation. The model that

they initiated consisted of artificial neurons where each neuron is characterized as either "on" or off," where an "on" occurs as a response to be stimulated by a sufficiently prescribed number of neighbor neurons. The legacy of their work was that they were able to show that any computable function was able to be performed by a network of connected neurons; and that all the logical connectives (and, or, not, etc.) could be realized by simple net structures [8].

Daniel Hebb developed a simple updating rule for changing the connection strengths between neurons that is still influential today (referred to as Hebbian learning). Marvin Minsky and Dean Edmonds built the first neural network computer in 1950, the Stochastic Neural Analog Reinforcement Calculator (SNARC) that built upon the foundations of Hebb's rule [8]. There were other pioneers and contributors to the birth of AI, but, Alan Turing remains the most influential and far-sighted.

2. Birth (1956)

The phrase "Artificial Intelligence" was coined by the researcher John McCarthy (known as the father of AI). The term was the result of a two-month workshop at Dartmouth College. McCarthy had invited ten other researchers to the workshop who were interested in automata theory, neural nets, and the study of intelligence. There were two researchers, Allen Newell and Herbert Simon, who brought about some of the more interesting ideas from the workshop. They had developed a program called "Logic Theorist," which they claimed was able to think "non-numerically." The program was able to prove various logical theorems (including a proof that was shorter than the original). McCarthy was asked years later about the coining of the phrase Artificial Intelligence and stated that "computational rationality" would have been more appropriate, but, AI has stuck to this day.

3. A Bright Future (Late 1950s–Late 1960s)

By the late 1950s, AI began to officially emerge as an accepted field of study. The "idea" of artificial intelligence led researchers to identify some specific tasks that were thought to require intelligence. The challenge then became to figure out how computers would be able to solve these tasks and problems. Researchers could obviously

not develop the HAL 9000 (fictional computer from Arthur C. Clarke's 2001: A Space Odyssey), but they could begin to tackle problems such as playing chess and checkers, proving mathematical theorems, answering relatively easy questions, and the classification of images. Researchers also worked to explain how human brains solved problems, to better model the problem solving of computer systems. This research brought about the intertwining of AI and cognitive psychology.

Much of the work done in this era also dealt with pattern recognition on two-dimensional material including text and pictures. A pioneer in pattern recognition and image processing was Russell Kirsch. In 1957, he built one of the first photograph scanners and he experimented with image processing programs. Most of the recognition methods at that time were dependent on matching a character against a template (if a character was more like an "A," the input was deemed to be an "A"). This era also saw much work in the realm of neural networks and the idea of "perceptrons" or computer elements based on the neural network of the human brain. Neural networks will be discussed in more depth later in Chapter III.

4. Adjusting Expectations (1970s)

Many of the lofty goals and predictions set by AI researchers in the early days had to be readjusted during the late 1960s and early 1970s. The main reason for this readjustment was that researchers soon learned that the scalability of AI from simple problems such as games and pattern recognition was not easily translatable into real world problems. Marvin Minsky and Seymour Papert, strong advocates of "Strong" AI (AI that is equal to or greater than human intelligence), published a paper that demonstrated that perceptrons were limited. This resulted in a decline in AI research funding. Neural network research would not reemerge as a field of study until the late 1980s [9]. Reflecting upon the problems experienced by researchers during this period, it becomes evident that the researchers were not setting unreachable goals, but that they were limited in processing power to solve the problems that they were defining.

5. AI Resurgence (The Rise of the Expert Systems [1980s])

The 1980s saw a rise in AI programs known as expert (knowledge-based) systems. The functions of the expert system were based on expertise derived from a large knowledge base (set of facts about the environment) and rules on how combinations of different elements combine for different outputs. The first successful expert system was DENDRAL, (which stood for "Dendritic Algorithm"). DENDRAL was able to identify chemical compounds based on readings from a spectrometer. Companies began to reinvest in AI research, hoping that the convenience of an expert system would give them the edge in the world of consulting.

6. Adjusting Expectations (Again) (Late 1980s–Early 1990s)

AI research again suffered financial setbacks during this short period, as researchers began to realize how intensive maintaining an expert system becomes. It became too difficult to update a system that was incapable of learning on its own. It was determined that expert systems were useful, but only in special circumstances.

7. AI as a Science (Early 1990s–Present)

Researchers have now begun to work more on building upon existing theories than proposing new theories (which often led to early boastful claims). The field of AI as a science means that researchers now also follow the scientific method of creating a hypothesis, experimenting, and reporting the analysis of the results (and their relative significance). The prevalence of the internet supports AI as a science because data and code are more easily shared and experiments can be replicated. The internet has also seen the development of search engines such as Google, Bing, and WolframAlpha, which all claim to use some form of AI algorithms in their searches. Some areas of research that have emerged during this period are in the following fields: data mining, Bayesian networks, and intelligent agents.

8. AI Research in the Future …

It is generally accepted that sensory systems alone (vision, audio, motion, etc.) cannot deliver a perfectly reliable interpretation about the environment. Therefore, the

fusion of sensory inputs will become a more widely researched area. The future will see AI replace various human functions more and more, including automatically fusing sensor inputs. The advances in AI clearly run parallel to the advances in microprocessor technology. Applications of AI might soon become more visible in such technologies as traffic analysis and routing, human behavior analysis, and home assistance (for elderly or disabled). Additionally, while it may seem trivial, computer games are pushing the envelope when it comes to the use of AI in games and the applicability of this technology will only continue to grow. As always, when the commercial marketplace sees the profitability of a technology, research is advanced even more.

Some researchers, and so-called "futurists," foresee a time in the "not too distant future" when "the singularity" (a time of "machines with greater than human intelligence") will become real [51]. Researchers must be careful with grandiose claims like these, for it was similar comments that led to the first disillusionment with AI. While it is important to have imagination when researching, researchers should remain realistic and work on problems that benefit and augment the actions of humans.

In this section, a quick history of AI was discussed; the next section will go into the basic concepts of AI.

C. THE BASIC CONCEPTS

For the purpose of this thesis, only the basic concepts of AI will be discussed (the reader is directed to the references for more advanced topics in AI).

1. Agents

An agent is anything (robot, software, etc.) that can be interpreted as being able to perceive its environment through sensors and act upon that environment through actuators. Figure 5 provides the basic conceptual visualization of what an agent is. Agents are systems capable of autonomous action in their environment in order to meet their specified objectives.

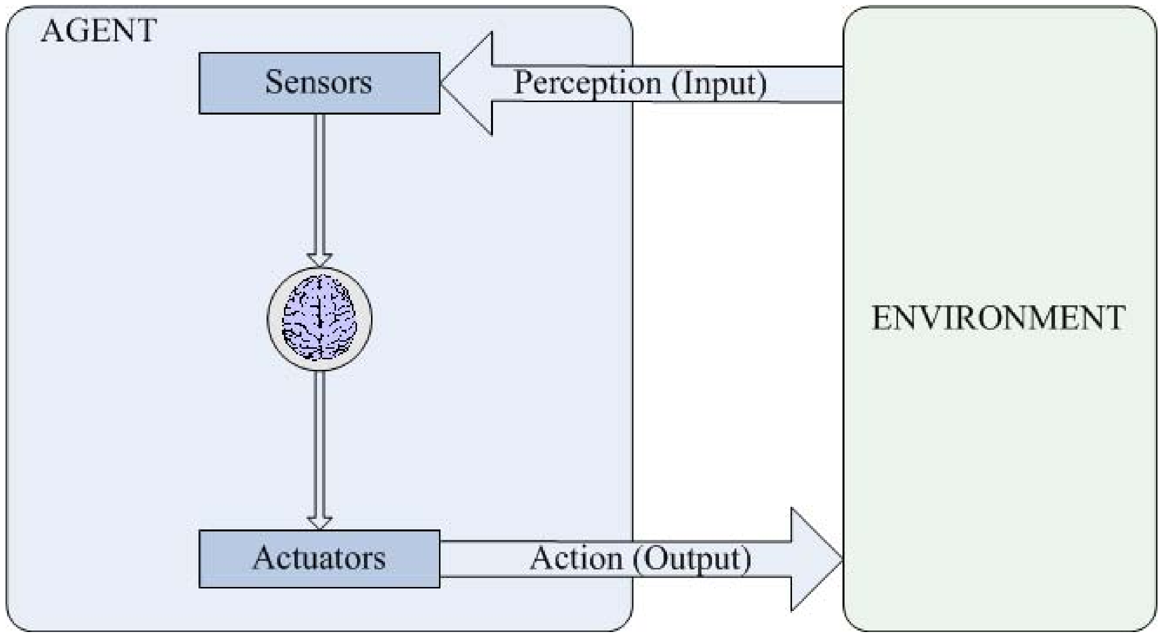

Figure 5. A Simple Agent. (After [8])

a. Examples of Agents

(1) Any control system. A classic (and simple) example of a control system is the basic thermostat. A thermostat has a sensor for detecting the temperature of the environment and an actuator that affects the environment temperature through turning the heating "on" or "off." The basic decision architecture of a thermostat encompasses the following rules: too cold—heat on; temperature at or above set value—heat off. Obviously, there are numerous examples of more complex control systems, but none as easily conceptualized as the thermostat system [9].

(2) Software daemons. A software daemon monitors a software environment and acts upon that software environment. For example, consider the background processes that run on Microsoft Outlook email. There are numerous software agents that run on this program, such as the agent that indicates to the user when a new email has arrived through continuous monitoring of the inbox and notification through the graphical user interface (GUI) that a new email has arrived. Additionally, the

"out of office" auto reply process can be viewed as a software agent; the agent monitors for new emails sent while the user is away and affects its environment by sending a reply to the sender [9].

b. Intelligent Agents

The information in this sub-section is contained in [9]. Applying the definition that was established earlier for intelligence, we can determine that an intelligent agent possesses the following three characteristics:

(1) Reactivity. Intelligent agents can perceive their environment and respond in a timely fashion to changes that might occur in order to satisfy their design objectives. It is important for an intelligent agent to find a balance between goal-directed and reactive behavior. It is desired that agents attempt to achieve goals systematically, but it is not desired for agents to achieve their goals by blindly executing procedures. It is necessary for an agent to be able to react to a new situation; to be continually reacting.

(2) Pro-activeness. Intelligent agents are able to exhibit goal-directed behavior by taking the initiatives in order to satisfy their design objectives. Building a proactive goal-directed agent is generally not very difficult. A simple function written in the computer language of *C* is an example. A program has various assumptions (pre-conditions) and the effect (post-condition) is the result if the assumptions are valid. The effect of the program is the goal that the author intended for the program.

(3) Social ability. Intelligent agents are capable of interacting with other agents (and humans if necessary) in order to satisfy their design objectives. The characteristic of "social ability" may seem trivial, but, the ability negotiate and cooperate between agents is very complex.

c. Agents and Expert systems

An expert system is one that is capable of solving problems or giving advice in some knowledge-rich domain. An expert system that is best explained by

describing the classic example of an expert system is MYCIN, which was intended to assist physicians in the treatment of blood infections in humans [8]. It worked by a process of interacting with a user in order to present the system with a number of facts, which the system then used to derive some conclusion (similar to a consultant, in that it did not directly operate on any humans, but affected the environment by providing expert knowledge). In this sense, expert systems (like MYCIN) are disembodied, meaning they do not interact directly with any environment; they get their information through a user, not via sensors (like an agent).

2. Agent Architectures

The information in this sub-section is contained in [8]. One of the jobs of AI is to design the agent program that implements the agent function, mapping inputs to outputs. The program will run on a computing device with physical sensors and actuators (the architecture).

a. Simple Reflex Agents

These agents select actions on the basis of the current input, ignoring the rest of the input history. These are considered condition-action ruled programs (if/then). Simple reflex agents are very simple and have very limited "intelligence." Figure 6 shows the general structure of the simple reflex agent, showing the importance of the condition-action rules on how the agent makes the connection from perception to action.

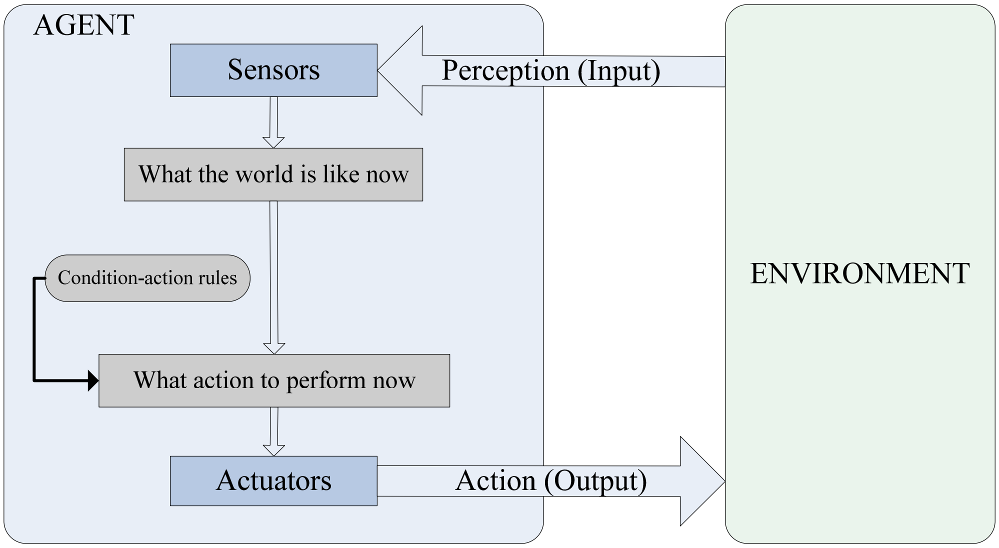

Figure 6. A Reflex Agent. (After [8])

b. Model-based Reflex Agents

When the environment is only partially observable (from noisy and inaccurate sensors or data is missing), an effective architecture is to maintain some type of internal state that depends on the input history and thereby reflects at least some of the unobserved aspects of the current state. This type of agent is considered a model-based agent (Figure 7). Figure 7 shows how the current perception of the environment is combined with the previous internal state to form an updated view of the environment.

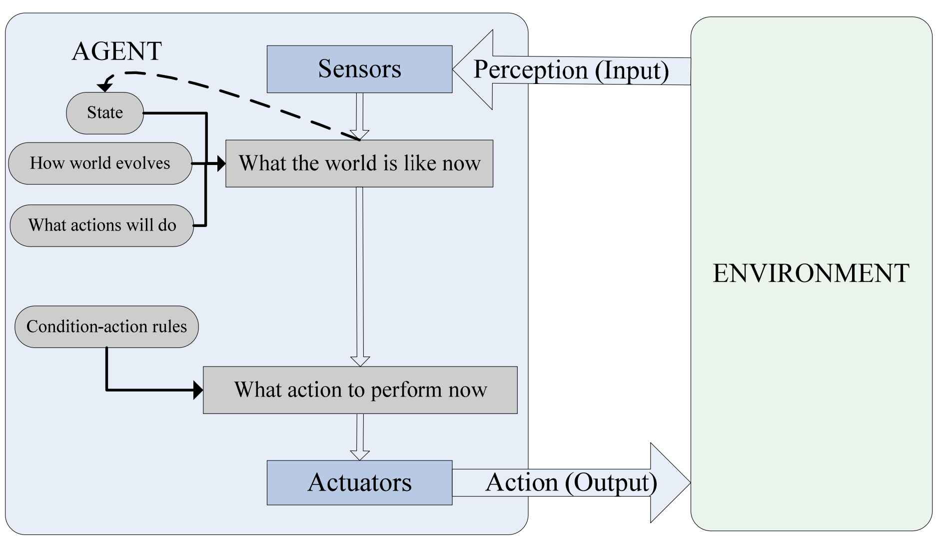

Figure 7. A Model-based Reflex Agent. (After [8])

c. Goal-based Reflex Agents

In addition to a current state description, the agent needs some sort of goal information that describes situations that are desirable. It keeps track of the world state as well as a set of goals it is trying to achieve, and choose an action that will eventually lead to the achievement of its goals. It may seem less efficient, but, it is more flexible because the knowledge that supports its decisions is represented explicitly and can be modified. Figure 8 is an abstraction of the goal-based reflex agent.

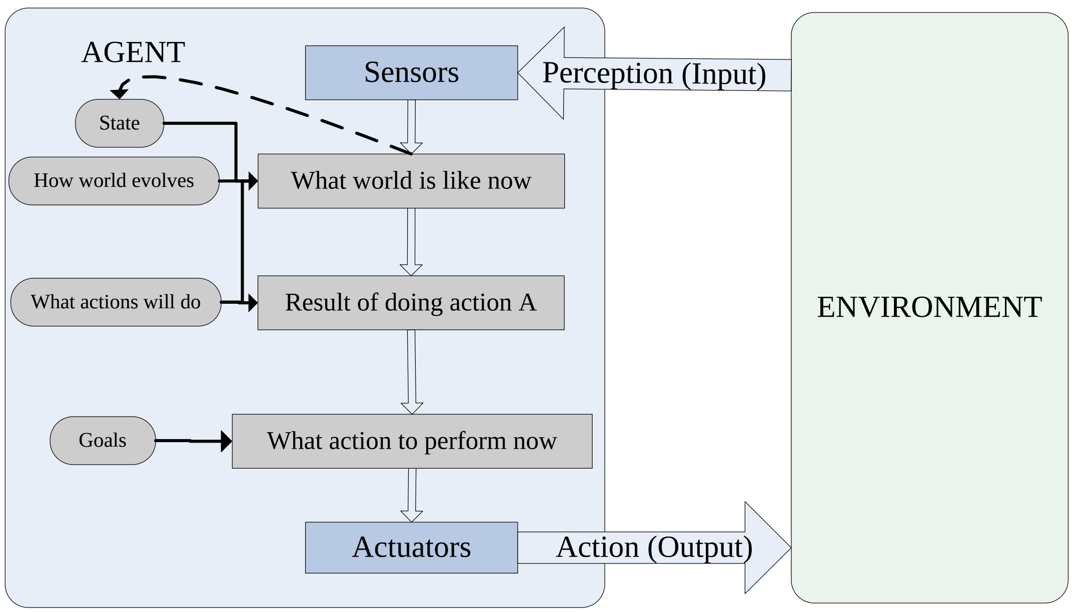

Figure 8. A Model-based, Goal Based Agent. (After [8])

3. Basic Algorithms (Search Paradigms)

Searching is an essential aspect of AI because problem solving in AI is basically a search. It is generally accepted that there are two search strategies: uninformed (no additional information given beyond what is already known) and informed (relevant information beyond the definition of the problem is utilized). Within each of these strategies are a number of search strategies, which make up the foundations for most advanced algorithms in use today.

These search algorithms can be used to augment and enhance military network-centric systems and applications. For the successful implementation of NCO and NCW,

it will be necessary to take these search algorithm concepts and use them for the automation of learning behaviors, intelligent states and features of interest in a network-centric environment. Modeling and implementing AI algorithms in network-centric applications typically requires a priori human expert knowledge to define behaviors, intelligent states, and features. In order to automate such defining of behaviors, intelligent states and features, it is necessary to utilize search algorithms to search network-centric environment historical data stored in databases. Such search algorithms could help learn and discover behaviors, intelligent states, and features of interest (inputs, outputs, algorithm details of each DIPR subsystem). Additionally, relative to multiagent systems, another example would be utilizing search algorithms to find objects in regions of interest (where each node in the search algorithm is a region of interest). Alternatively, regions of interest could be thought of as databases and other knowledge base systems that store data.

a. Uninformed Search Strategies

(1) Breadth-first search. Breadth-first search (BFS) is one of the simplest uninformed search strategies. In BFS, the graph (or problem space) is expanded from the root (or top) node. The top node is searched for the solution and then the nodes below are searched for the solution. BFS utilizes the first-in-first-out (FIFO) queue, which assures that nodes that are visited first are expanded first [8]. Figure 9 shows an abstraction of the search order used in BFS (numbers indicate the order in which nodes are expanded). An example of BFS can be visualized in how a designer sets up an algorithm to search a database for a particular piece of information (node). In the case of BFS, it would be used for a program in which data is added to the database sequentially .

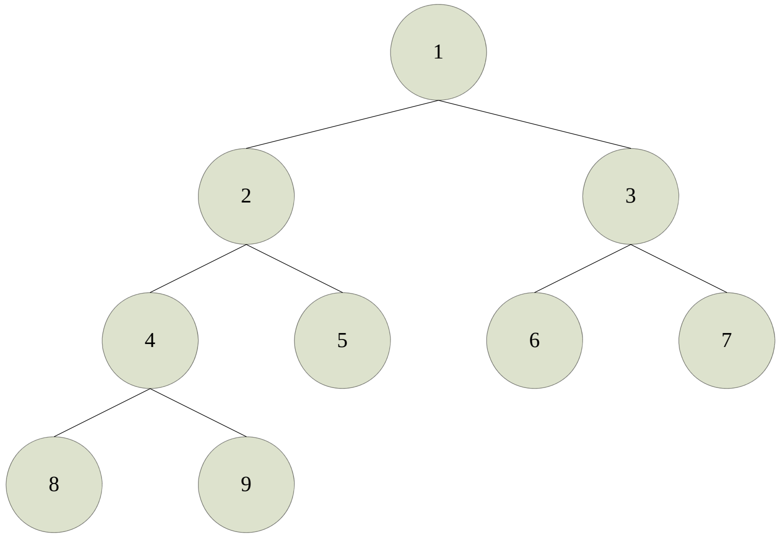

Figure 9. Breadth-first Search Algorithm.

(2) Uniform-cost search. BFS works great when the cost (power used, memory needed, number of computations, etc.) of each step is the same. Uniform-cost search (UCS) is concerned with the total cost of each step made down the graph towards the goal [8]. To ensure the lowest total cost is followed, UCS expands the node with the lowest path cost, illustrated by Figure 10 (path costs are depicted by the numbers on the connecting lines). This can often lead to a much longer path being followed to obtain the solution. Twenty steps down paths at a cost of two (total cost of forty) is much more than one step at a cost of three (total cost of three). An obvious drawback with using UCS is that the goal or solution may never be discovered. UCS utilizes a priority queue for searching (the node down the path with the least cost will get explored). An example of where UCS search algorithms might be used would be where the designer assigns different weights to a certain attribute, for example facial attributes might be weighted more than voice attributes in determining the identity of an individual.

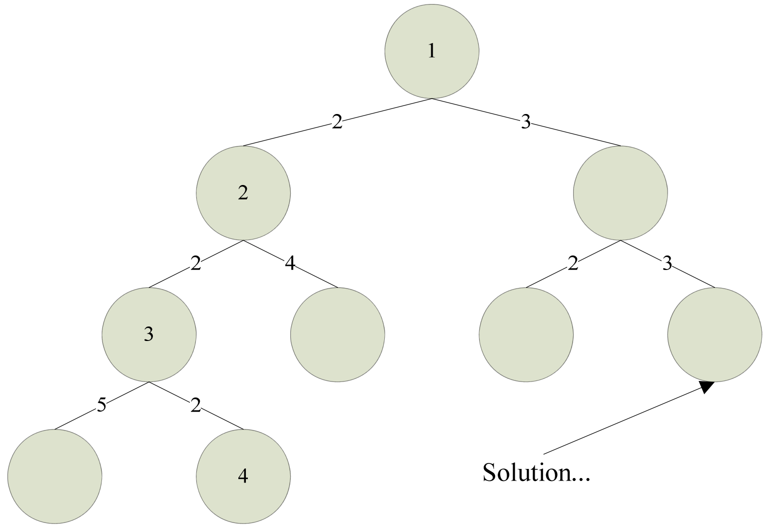

Figure 10. Uniform-cost Search Algorithm.

(3) Depth-first search. Depth-first search (DFS) searches the graph by expanding each branch to the deepest node, as depicted in Figure 11. Where BFS utilized the FIFO queue, DFS is based on the last-in-first-out (LIFO) principle where the last item placed on the top of the stack is the first item to be removed [8]. A drawback from using DFS is that the algorithm can search down a very long branch without ever coming to the solution node. An example of DFS, like BFS, can be visualized in how the setup of a database. DFS would be used for a program in which data is added to the database cumulatively (i.e., information is added to the top of the stack).

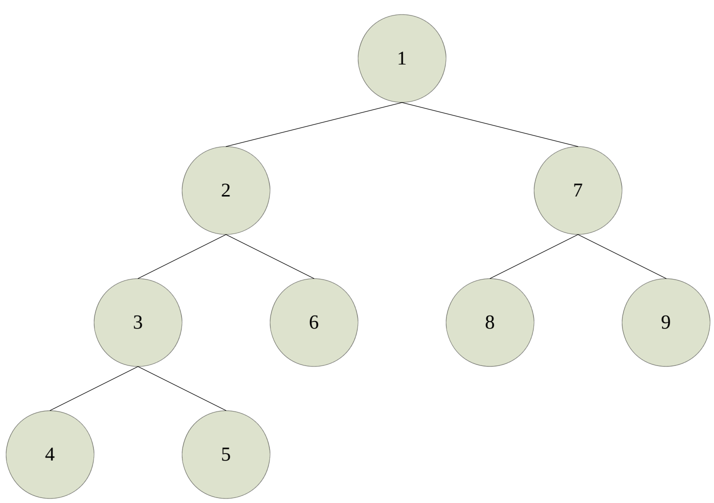

Figure 11. Depth-first Search Algorithm.

(4) Depth-limited search. Depth-limited search (DLS) attempts to prevent the drawback of the DFS mentioned earlier. It modifies DFS by minimizing the depth that the search algorithm may search [8]. An obvious problem associated with DLS is if the depth selected is above the depth of the solution as illustrated by Figure 12 (with a predefined search depth of 2). An example of when DLS might be used is when the designer needs to optimize the amount of time it takes to find a goal (i.e., the time used is weighted more than the actual discover of the goal).

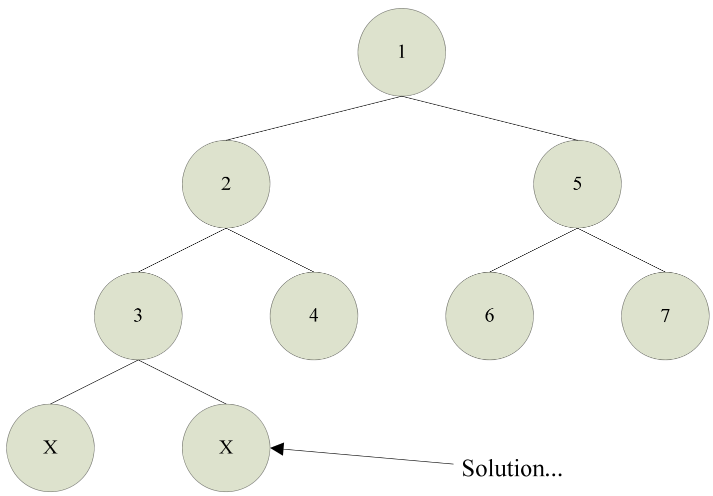

Figure 12. Depth-limited Search Algorithm.

(5) Iterative deepening search. Iterative deepening search (IDS) is a variation of DLS. It combines the features of DFS with that of BFS [8]. IDS operates by performing DLS searches with iterative depths until the solution is found, as shown by Figure 13. The depth chosen begins at one and iteratively increments one depth level. Generally, IDS is the preferred uninformed search when there is a large search space and the depth of the solution is unknown, for example a computer chess game program.

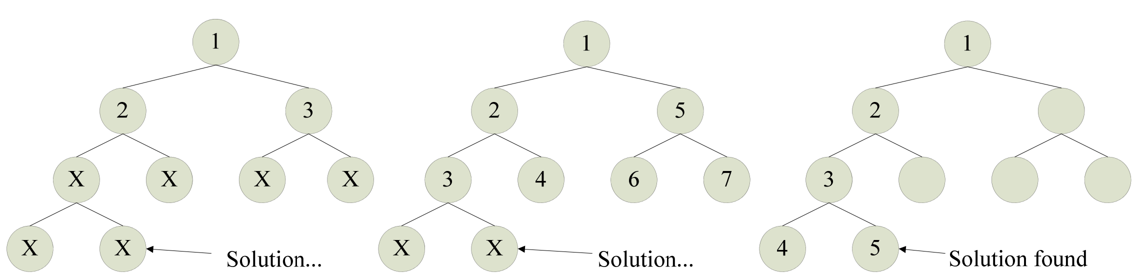

Figure 13. Iterative Deepening Search Algorithm.

b. Informed (Heuristic) Search Strategies

(1) Greedy Best-first search. The greedy best-first search (greedy search) evaluates the search space based on a heuristic function. A heuristic function, h(n), is the "*estimated* cost of the cheapest path from node *n* to a goal node [8]," where "node *n*" is the node in the search space the algorithm is at that instant and the "goal node" is the location of the solution. Additionally, the term "cheapest path" is used to show that it is desired to find the optimal or cheapest path, which can be defined for whatever the search state is (i.e., minimize: power, time, distance, etc.). Defining the evaluation function as f(n), the greedy search is simply f(n) = h(n). A classic example of the greedy search is the "straight-line distance" (SLD) heuristic. The SLD heuristic assumes that the node with the smallest "straight line distance" to the goal will provide the optimal solution. This assumption does not account for the *actual* cost to reach each node (the heuristic is an estimation and the decision is based only on the SLD). This example of the SLD heuristic is best explained by examining the example presented in Figure 14.

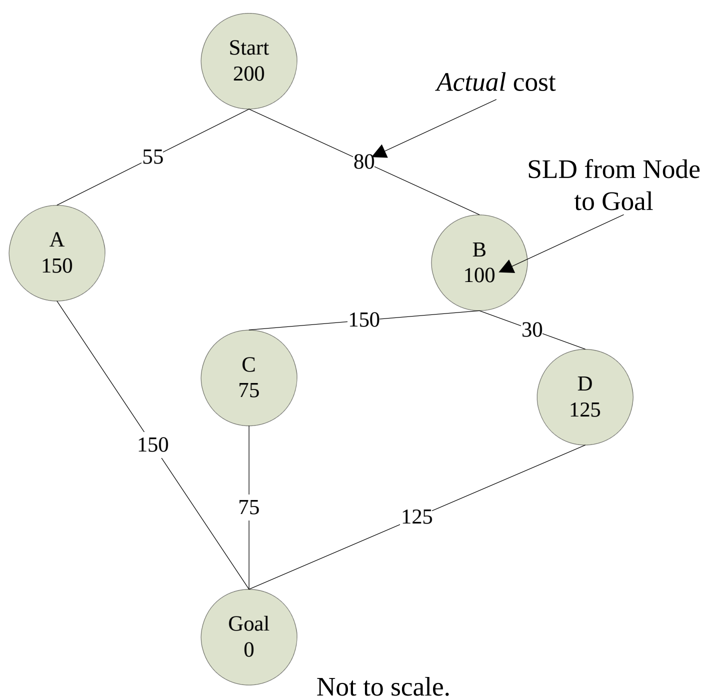

Figure 14. Example of Greedy Search Algorithm.

In Figure 14, each node is shown with the respective SLD values underneath the node identifier (letter) and the numbers on the paths (lines) are the cost to travel to the respective node. It can be seen that there are three separate paths that can be followed from Start to Goal: A; B-C; and B-D. Based on the greedy search algorithm for SLD, the B-C path would be selected with a total cost of 305 (80 + 150 + 75). This illustrates the obvious fault associated with the greedy search algorithm: the heuristic chosen does not always lead to the most optimal solution (which in this case would have been just to go to node A then the Goal at a cost of only 205). A real world example of a greedy search algorithm would be an autonomous robot that has a priori knowledge of its environment and makes it decision of how to travel to the goal based on distance traveled, the path the robot travels is based on the straight line distance to the goal, where the distance to the goal is correlated to the power consumed by the robot (implied sub-goal is to minimize power consumption).

(2) A* search. A* ("A-star") search, like the greedy search algorithm, evaluates a search space using a heuristic function as well as taking into account the cost function of getting from the current node to the next one (g(n)). So, the A* function is defined as f(n) = h(n) + g(n), where g(n) is the cost to get to a node and h(n) is the cost to get from the node to the goal [8]. The A* search is best visualized by re-examining the search space represented in Figure 14, now shown in Figure 15. In Figure 15, when the cost function is taken into account the path selected is simply Start-A-Goal, as A has the optimal solution for f(n) (a total cost of 205). The other two paths, Start-B-C-Goal and Start-B-D-Goal, result in solutions of 305 and 235, respectively. Taking the previous example of the robot into consideration, where the robot did not account for traveling around corners or other obstacles, the A* search optimizes not just the distance to the goal, but the distance traveled on each leg of the journey to the goal.

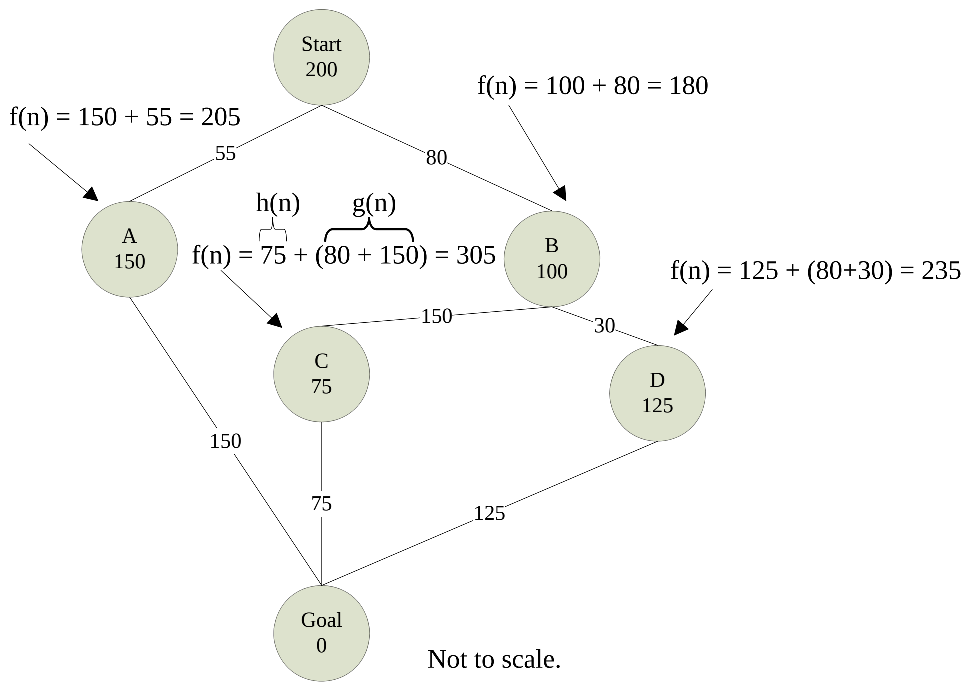

Figure 15. Example of A* Algorithm.

(3) Others. There are numerous other informed search methods that have been developed, such as hill-climbing search and simulated annealing (33). Hill climbing is an iterative improvement algorithm that is similar to the greedy search, except that there is no backtracking. At each step of the search, a single node is selected to examine. The decision for which node to select is that it “improves” upon the current state (i.e., climbs up the hill). The problem with hill-climbing is that it can lead to the solution of a local maximum, rather than the global maximum (optimal solution), as can be seen in Figure 16.

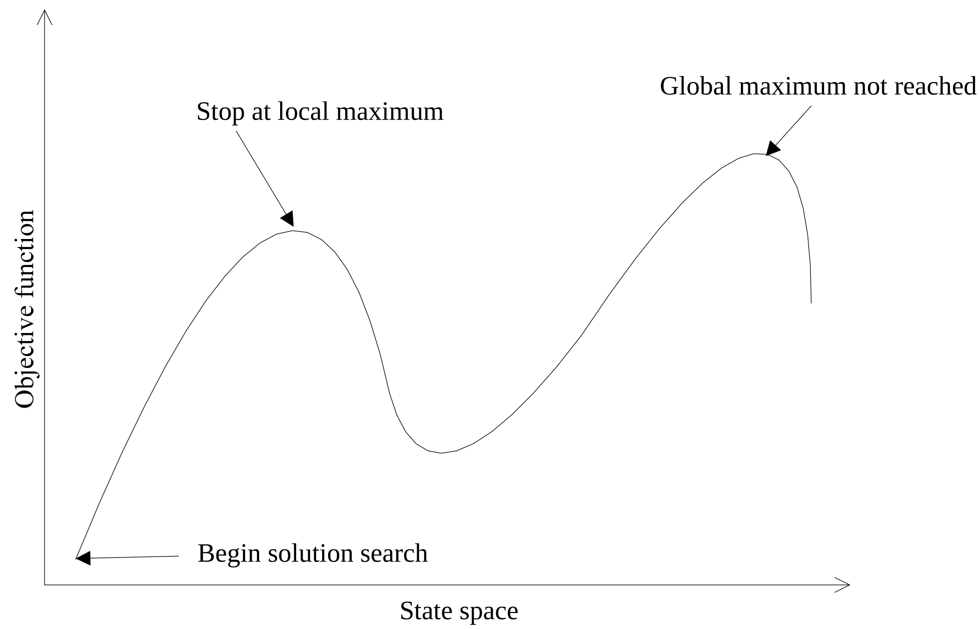

Figure 16. Hill-climbing Search.

As the name implies, simulated annealing mimics the process of annealing (gradual heating and cooling of metal to obtain a higher quality metal). Since hill-climbing will never go "downhill," the optimal solution might be missed, as it was in Figure 16. But, by introducing an element of randomness (i.e., randomly changing starting positions for the search), the chance of getting to the global maximum is greatly increased.

c. *Search Strategies Summary*

While it may seem that the analysis of search algorithms seem tedious, they form the foundation upon which an agent "recollects" information. Just like human intelligence is dependent upon individuals retrieving information from memory (in very unique search algorithms), artificial intelligence is dependent upon how information is retrieved from databases. Search algorithms are the cornerstone of AI and therefore a necessary field to be understood designing the various classifiers used by smart sensor networks.

4. Low-level Classifiers for Object Detection

Referring to the detection function of the DIPR architecture provides an example of what a low-level classifier is. After the search algorithms are performed to locate an object of interest in the network-centric environment and the raw sensor data is extracted, the low level classifiers extract the actual object spatial-temporal features. There are various types of algorithms that can extract features from raw sensor data. Some of these are described below.

a. Computer Vision Algorithms

These algorithms include face recognition, posture recognition, hand gesture, camera calibration, object recognition and character (letter/number) recognition [10, 11]. An example of a camera calibration algorithm will be discussed (Figures 17 and 18). This example is based on the calibration of WiLife cameras using MATLAB software in support of the SE4900 project, Watchman, conducted by NCSE students [57]. In this calibration, an image was obtained from a camera and processed and analyzed to obtain a matrix of centroids for pre-determined points (on a 3 ft by 3 ft spacing) (Figure 17). The image was read in to the program so that it could be manipulated by MATLAB. The image went through pre-processing to include converting to grayscale, sharpening and reducing background noise, and converting to black and white. A morphological operation was then performed on the image, where a structuring element was defined to find the desired shapes (in this case it was a disk). The image was then cleaned up some more by filling in inconsequential "holes." The program then defined the criteria that it would be analyzing (it was assigning centroids to objects with a specified area). Finally, the program ran a function loop on the image, which examined each object in the image (white blob) and determined if it met the criteria and assigned the centroid points to each object, outputting a matrix of centroid points, visualized by the blue crosses (Figure 18). This is an example of using AI low-level classifiers to automate the defining of regions of interest [57].

Figure 17. Pre-Analyzed Image (Raw Data).

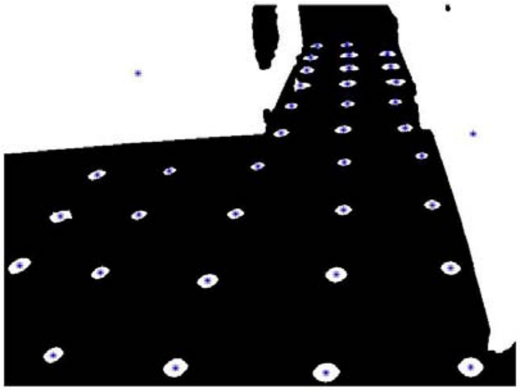

Figure 18. Analyzed Image (Interpreted Data).

b. Audio Algorithms

These algorithms include speech recognition and voice recognition [12], [13]. These algorithms are used in interpreting what an individual is saying and who is saying it. An everyday example of where speech recognition is seen (or heard) is in recent cell phone technology, and the use of the "call" command (e.g., "call home"). Also, the voice recognition algorithms are seen as being able to add an extra element of security (since each voice has a unique characteristic).

c. Signal Processing Algorithms

These algorithms include signal detection and signal classification [14]. These algorithms might be seen on various platforms and utilized by different sensors (such as radar and sonar). The algorithms could be applied to classifying a certain track or target as another type of platform.

d. Other Statistical Algorithms.

These algorithms include Bayesian, principal component analysis, adaptive boosting (AdaBoost), and neural networks (to be discussed later) [15], [16].

5. High-level Classifiers

An example of what a high-level classifier is illustrated by the prediction function DIPR architecture. After the object recognition is completed, sequences over time and space are initiated. The functions performed include spatial-temporal pattern recognition and behavior classification analysis. Examples of these high-level classifiers are hidden Markov models [13], weighted finite state machines [4], context free grammars [17], finite state machines [17], other syntactical classifiers, and other state based classifiers [18].

6. High-level Classifiers Used to Fix Low-level Classifiers

As discussed, low-level classifiers perform the function of detecting a feature and the high-level classifiers predict some type of result from the sequences of the objects. A new way of examining these high- and low-level classifiers is to use a high-level classifier to "fix" (enhance) low-level classifiers. The intention is to reduce the amount of tuning and re-training required, making low-level classification more accurate. Instead, high-level classifiers can intelligently correct errors of low-level classifiers without retraining low-level classifiers. This is important, since artificial intelligent network-centric system of systems depend on the accuracy of the low-level classifiers that comprise the bottom-up approach to NCSE [18].

In this section, the basic concepts of AI (agents and algorithms) were presented; the next section provides a brief summary of the chapter.

D. SUMMARY

After reviewing the concepts presented in this chapter, it becomes clear that there is much involved in the study of AI, from the rich history of AI, to the concepts of agents and search strategies. A network-centric systems engineer must be an expert in network-centric systems; it should not be expected that they be an expert in any of these topics (a network-centric systems engineer would interact with AI experts), but, it is beneficial (and encouraged) that a basic understanding and appreciation for the introductory topics be had. With a basic background in AI, it becomes easier for the engineer to be more equipped at understanding the concept of distributed AI and multiagent systems, the building blocks of a successful smart sensor network and the key to implementing of NCW.

In this chapter, a brief background of AI was presented, along with the basic concepts of AI; the next chapter provides a discussion of distributed artificial intelligence and multiagent systems.

THIS PAGE INTENTIONALLY LEFT BLANK

III. DISTRIBUTED ARTIFICIAL INTELLIGENCE AND MULTIAGENT SYSTEMS

A. OVERVIEW

Distributed Artificial Intelligence (DAI) is associated with systems of interacting agents (a multiagent system) that are working towards a common goal. DAI and multiagent systems find a natural niche in the world of NCW. The application of DAI through the multiagent systems, embodied in a smart sensor network, is the quintessential network-centric system (of systems). Today’s joint forces are composed of a large number of subsystems with different natures, with different functions and composed of human operators (distributed across the battle environment). Through DAI, these various subsystems that utilize sensor networks will be able to be fused for more efficient information sharing. By becoming familiar with the concepts of DAI and multiagent systems (MAS), the network-centric systems engineer will have the tools necessary to implement a smart sensor network. DIPR and MAS are considered applications of a system of systems. Table 1 differentiates a system from a system of systems. In Table 1, the attributes under the System of Systems heading can be related, and are applicable, to DAI and, therefore, DIPR and MAS.

Element	System	System of Systems (SoS)
Autonomy	Autonomy is ceded by parts in order to grant autonomy to the system.	Autonomy is exercised by constituent systems in order to fulfill the purpose of the SoS.
Belonging	Parts are akin to family members; they did not choose themselves but came from parents. Belonging of parts is in their nature.	Constituent systems choose to belong on a cost/benefits basis; also in order to cause greater fulfillment of their own purposes, and because of belief in the SoS supra purpose.
Connectivity	Prescient design, along with parts, with high connectivity hidden in elements, and a minimum connectivity among major subsystems.	Dynamically supplied by constituent systems with every possibility of myriad connections between constituent systems, possibly via a net-centric architecture, to enhance SoS capability.
Diversity	Managed i.e., reduced or minimized by modular hierarchy; parts' diversity encapsulated to create a known discrete module whose nature is to project simplicity into the next level of the hierarchy.	Increased diversity in SoS capability achieved by rereleased autonomy, committed belonging, and open connectivity.
Emergence	Foreseen, both good and bad behavior and designed in or tested out as appropriate.	Enhanced by deliberately not being foreseen, though its crucial importance is, and by creating an emergence capability climate, that will support early detection and elimination of bad behaviors.

Table 1. Systems vs. Systems of Systems (AI vs. DAI). (After [19])

In this chapter, the following topics will be covered:

- Why distributed AI?
- What is an MAS?
- DIPR as MAS

In this section, an overview of the chapter was discussed; the next section discusses the need for distributed artificial intelligence.

B. WHY DISTRIBUTED AI?

A system is better suited to act in the desired manner (towards achievement of a goal or mission) if the actions are efficiently coordinated. The need to distribute intelligence is therefore explained by the following principles [20]:

- It is necessary to automate detection, identification, prediction and reaction to globally distributed potential terror threats.
- As the number of sensors rapidly increases in numbers and types, there will never be enough: humans, intelligent centers, or bandwidth. Once mobile, there will never be enough: bandwidth, power, or weight. The unmanned world has to be automated with AI software system of systems.
- Problems encountered by the military are inherently physically distributed.
- Networking of forces (and sensors) compels a distributed view.
- The complexity of problems faced by the military mostly requires a distributed approach to solving. For example, it is much more difficult to "track the movements of a person of interest" then to "report all visual sightings of this person of interest automatically" (and then to be analyzed at a higher level).
- It is easier for a distributed system to adapt to changes in the environment (or commanders' requirements). For instance, it is much easier to modify the above example by just adding another requirement to the distributed sub-systems then to say, "in addition to tracking the movement of this person of interest, track the movement of another."

Goshorn et al.[4] also make profound arguments for the necessity of DAI. It is a fact of modern life and warfare that the U.S. is in a global struggle against violent extremists. These threats cannot be ignored, and will not be going away any time soon. To mitigate any effect these entities might have on the way of life in the U.S., Goshorn identifies the need to "automate: detection, identification, prediction and reaction" of these globally distributed threats, implemented through a network-centric AI system of systems (i.e., DAI) [4].

A brief discussion of the proposed standard system of "Detect, Identify, Predict and React" (DIPR) is presented below along with a visual overview of the system (Figure 19) [4]. The environment is shown at the center of the diagram to show that the

environment is essentially what the whole system revolves around. Information is obtained from the environment and analyzed through DIPR to result in an effect on the environment.

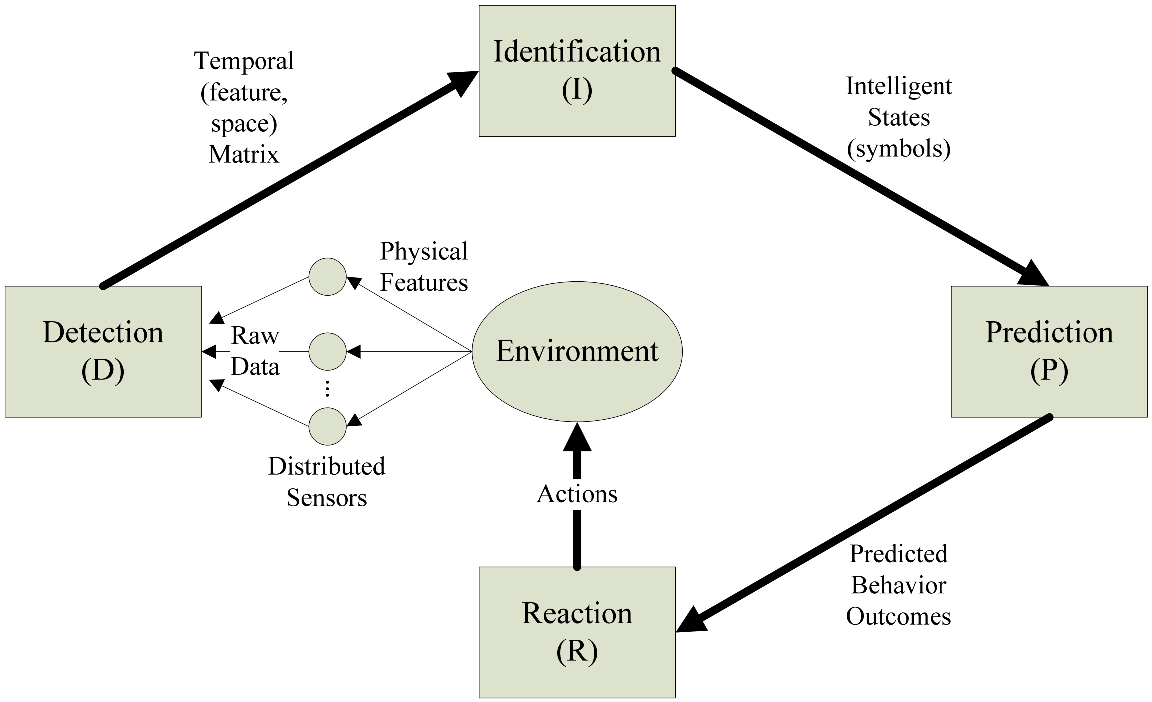

Figure 19. DIPR System. (After [4])

1. Detection

The detection subsystem processes the raw data input from the sensors. The output of this subsystem is a spatial-temporal feature matrix. It is important that adequate attention be given to this subsystem when designing a network-centric DIPR system since this is where the data begins to be analyzed. The engineer must build a detection architecture that not only takes into account the desired feature to be extracted from the data (i.e., visual data), but also the time and space of data extraction for future fusion and correlation of data. It is worth noting that the detection process may be done either locally at the sensor or at a node processing system, thus enabling DAI. Both methods have their benefits; the benefit of detecting locally is that it reduces bandwidth used and processing power used at the node; the benefit of detecting at the node system is that there is generally more processing power at the computer. A network-centric systems

engineer must be able to weigh the pros and cons of both methods. In general, the keys to the detection sub-system are the “low-level classifiers” (what, when, where), filters and methods for feature detection (Figure 20).

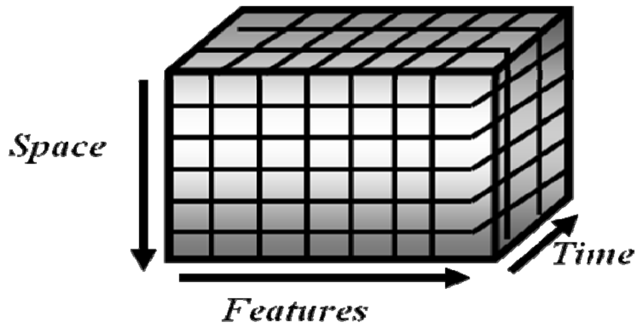

Figure 20. Spatial-Temporal Feature Matrix. (From [4])

2. Identification

The identification subsystem fuses elements of the spatial-temporal matrix output from the detection subsystem to identify the “intelligent state” (symbol) of the observed environment at a time step. The intelligent states are a result of fusing the elements of the spatial-temporal matrix in accordance with pre-set rules (usually defined by the user) (e.g., the features of an image were identified as a person). A corresponding (arbitrary) symbol is assigned to each desired fusion of feature and spatial-temporal attributes for further analysis by the prediction subsystem. Refer to Figure 21 for the workings of the identification function. The identification function is essential to DAI because it automates the fusion of extracted features from distributed multi-modal sensors.

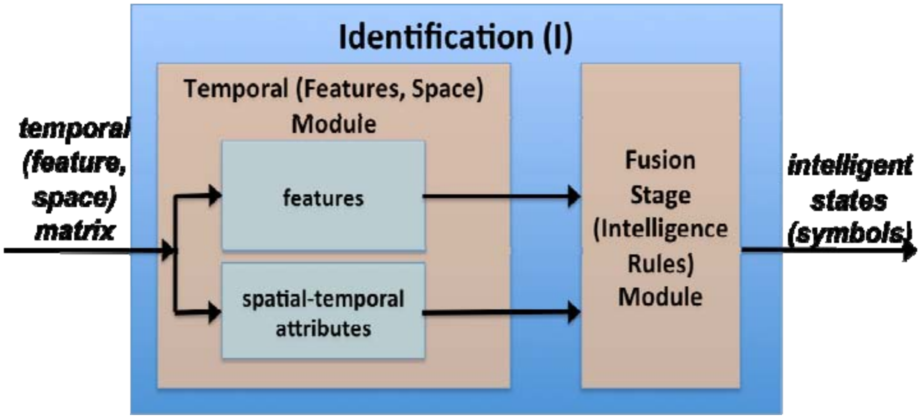

Figure 21. Identification Subsystem. (From [4])

3. Prediction

The prediction subsystem (Figure 22) carries out the high-level classification of the information flow in a network. The output of the prediction subsystem is based on the sequence of such symbols outputted from the identification subsystem. A behavior outcome based on the sequence of symbols is classified using a higher level AI classifier. For example, consider the following unique symbol sequences: aaaaa; ababab; abcabc, where symbol sequences are sequential intelligent states of the network, and must be classified as normal, abnormal, or unknown. These symbols are all three unique symbol sequences associated with a specific behavior and may be labeled as such: aaaaa is associated with normal behavior; ababab is associated with abnormal behavior; and abcbc is associated with unknown behavior, as it fits no known behavior. Predicted outcomes from classified behaviors are then inferred (e.g., normal behavior infers "everything is okay"; abnormal behavior infers "potential terrorist left luggage"; and unknown behavior infers "not sure what will happen"). These inferred predicted behaviors are then inputs to the reaction subsystem.

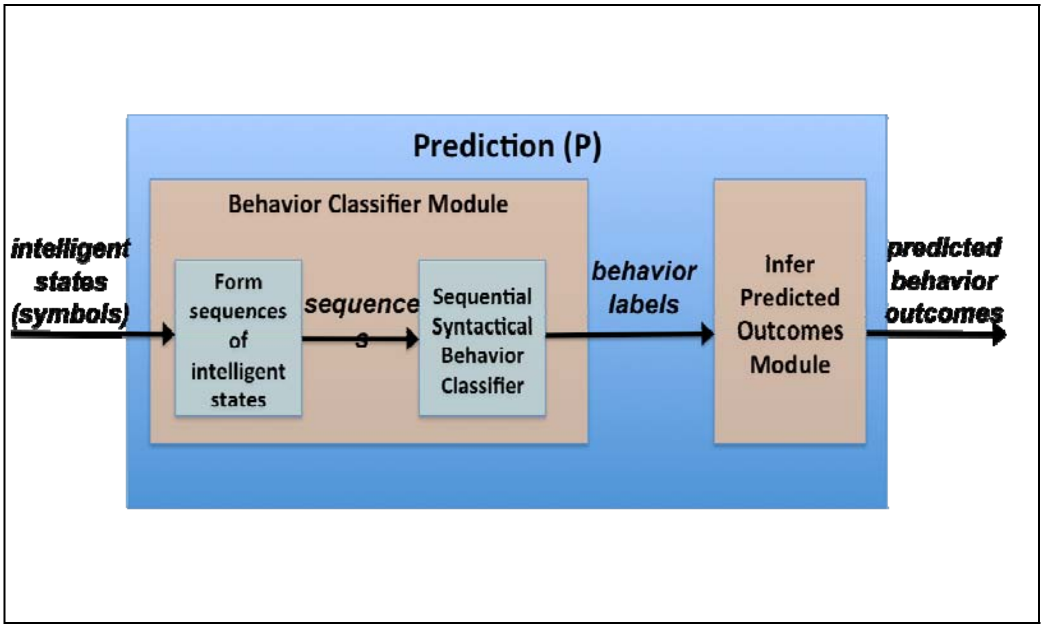

Figure 22. Prediction Subsystem. (From [4])

4. Reaction

The reaction subsystem develops actions to respond to the predicted outcomes output by the prediction subsystem. These reactions are usually pre-defined and are application dependent. For example, consider again the inferred predicted behaviors from earlier: "everything is okay" may have an action "do nothing"; "potential terrorist left luggage" may have an action of "sound alarm," and "not sure what will happen" may have an action of "notify user" to update system. The reaction function is essential to NCW and NCO, as it enables global action on a network as a response to the information collected from the network.

5. The Importance of DAI

The significance and promise of DAI began to be seen in the early 1990s. Most early work done in DAI was geared towards sensory networks, such as air traffic control, road traffic control, and robotic systems; basically, any application that required distributed interpretation and distributed planning by means of smart sensors. The positive aspects of performance, reliability, modularity and resource sharing began to be associated with DAI. An early "success story" associated with DAI was the "Pilot's

Associate” program, a five-year ARPA-funded program that demonstrated the application of DAI to help pilots of advance fighter aircraft. The goal was to provide the pilot with “enhanced situational awareness by sorting and prioritizing data, analyzing sensor and aircraft system data, distilling the data into relevant information, and managing the presentation of that information to the pilot [21].” Recommendations on actions to achieve the mission were then presented for the pilot’s action.

The DAI approach can be used to manage the complex nature of modern military applications. An important advantage associated with DAI is the ability to integrate existing standalone knowledge-based systems. This is important because applications and sensors used in the military are often developed in an ad hoc fashion. Ideally, if, a network is not designed with a common protocol, then DAI can take that into account when developing the various agents of the MAS.

In this section, the need for distributed artificial intelligence was discussed; the next section provides a discussion of what a multiagent system is.

C. WHAT IS A MULTIAGENT SYSTEM?

A multiagent system (MAS) can be considered a system in which several interacting intelligent agents pursue a set of individually-held goals, or perform separate tasks for the accomplishment of a common goal(s) [22]. Reference [23] states that “MAS approaches set out from an interaction-centered perspective: it takes a bottom-up approach and studies the properties that emerge from the interactions in such manifold collections of agents.”

The design of a single agent system, although not easy, is much less complicated than the design of an MAS. MASs require additional considerations, such as communication mechanisms, enhanced knowledge of the environment, and “societal” issues (which agent is assigned to a particular task) [24]. Additionally, the agents’ capabilities, availability and intended purpose must be considered when designing an MAS to solve a problem. Search algorithms that were used for single agent systems must be tailored and scaled for an MAS [9].

Characteristics of an MAS, identified in [25], are below:

1. Agent Design

Various agents that make up an MAS may be designed in different ways (e.g., different sensors) or be exact duplicates of each other. Therefore, agent design may be considered to be either heterogeneous or homogenous.

2. Environment

Agents can either be in a static or dynamic environment. Single agents are usually designed for use in a static environment and have associated AI techniques to deal with that environment. The mere presence of more than one agent automatically makes the environment of an MAS dynamic.

3. Perception

The perceived information that is received by the agents (through sensors) is obviously distributed in the sense that the agents will observe data hat differs spatially (different places), temporally (different times) or semantically (different interpretation of data). This makes the world partially observable to each agent.

4. Control

Control in an MAS is generally decentralized, meaning that the decision making of each agent lies mainly within the agent itself.

5. Knowledge

How well an agent's knowledge of the current world state can differ between agents in an MAS. In an MAS, each agent must generally consider the knowledge of each other agent in its decision making.

6. Communication

Communication between agents is required if agents are to interact within an MAS. Each agent in an MAS is potentially a sender and a receiver and there must be a standard language (or protocol) to facilitate agent communication.

Some key benefits of using an MAS in describing a large system are [25]:

- Speed and efficiency of parallel computation
- Robustness and reliability (system is not completely degraded when one or more agents fail)
- Scalability and flexibility allow for easy additions of new agents to the system
- Low cost, high reward
- Reusable and modular

In this section, the basic concepts of an MAS were presented; the next section compares DIPR and MAS.

D. DIPR AS MAS

While DIPR and MAS might seem to be not related at all, they are in actuality very closely related. DIPR and MAS are both AI system of systems approaches to automation in network-centric systems. In fact, MAS can be seen as existing in each element of DIPR, in the sense that each function is an agent that performs a specific function in its respective environment and then affects that environment (output). Figure 23 is a visualization of this concept.

Figure 23 presents each DIPR function as separate system, connected through an environment of interfaces. The agents of each system (feature detection, fusing, etc.) serve as facilitators to the accomplishment of the respective system goals (output matrix, predicted behaviors, etc.). These system goals are necessary and all provide accomplishment of the goal of the desired application (take away one system and the overall SoS cannot function). This is an example of a system where the whole is greater than the sum of the parts.

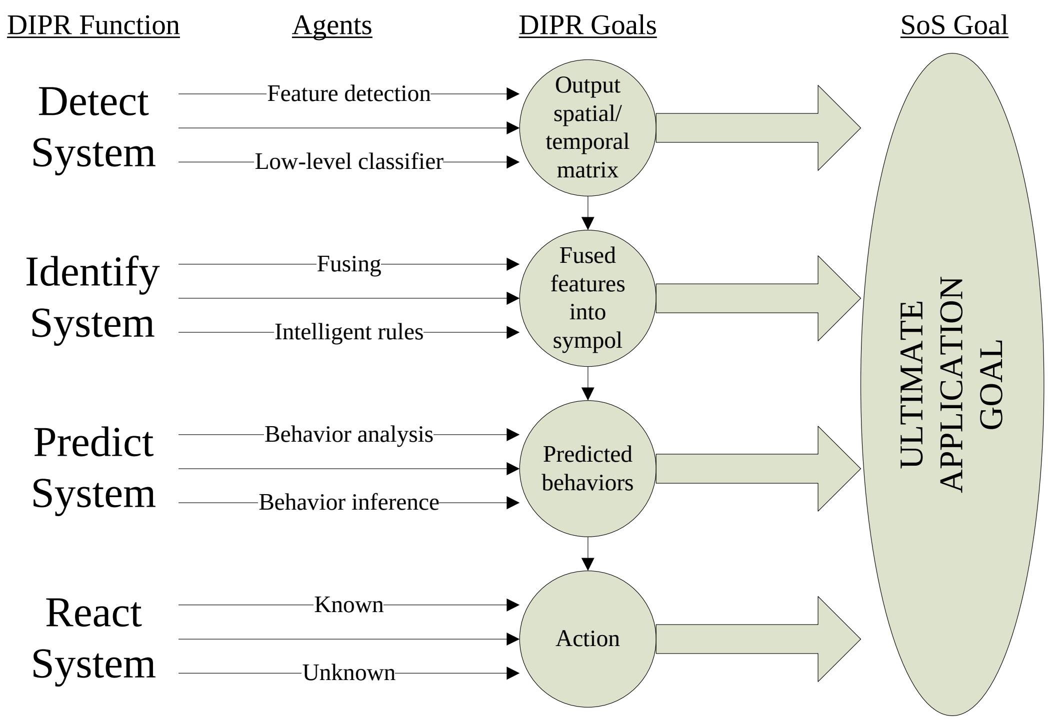

Figure 23. DIPR as MAS.

In this chapter, distributed artificial intelligence and multiagent systems were discussed; the next chapter discusses sensor networks.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. SENSOR NETWORKS

A. OVERVIEW

Sensors are the primary sources of information for systems (a priori or updated data input by user are other sources). Information obtained via a single sensor can be non-uniform, sporadic, misleading, incomplete, and flawed [30]. Creating a network of smart sensors reduces these information shortcomings and adds the element of data redundancy and a more inclusive, true, and consistent picture of the environment is obtained (assuming data is accurate) [31]. Sensor networks are the cornerstone of NCW and NCO, as without a sensor network, the full implementation of NCW cannot be realized. The following topics will be covered in this chapter:

- Sensor Overview
- Data Fusion
- Advanced Algorithms

In this section, an overview of the chapter was discussed; the next section serves as an introduction to sensor networks.

B. SENSOR OVERVIEW

As “smart” sensors become more and more accessible (in both cost and technology available), the end users’ requirements list is naturally going to grow along with it. Three main factors are identified in [26] that have contributed to the recent expansion of smart sensing: decreasing sensor cost, implanted microcontrollers, microprocessors and analog-to-digital converters; the proliferation of networking and diagnostic software; and the push for sensor interface standards and ontologies [26]. This is to be expected when one considers the implications of Moore’s Law, which states that the number of transistors per integrated circuit will grow exponentially over time (depicted graphically, specific to Intel® microprocessors, in Figure 24).

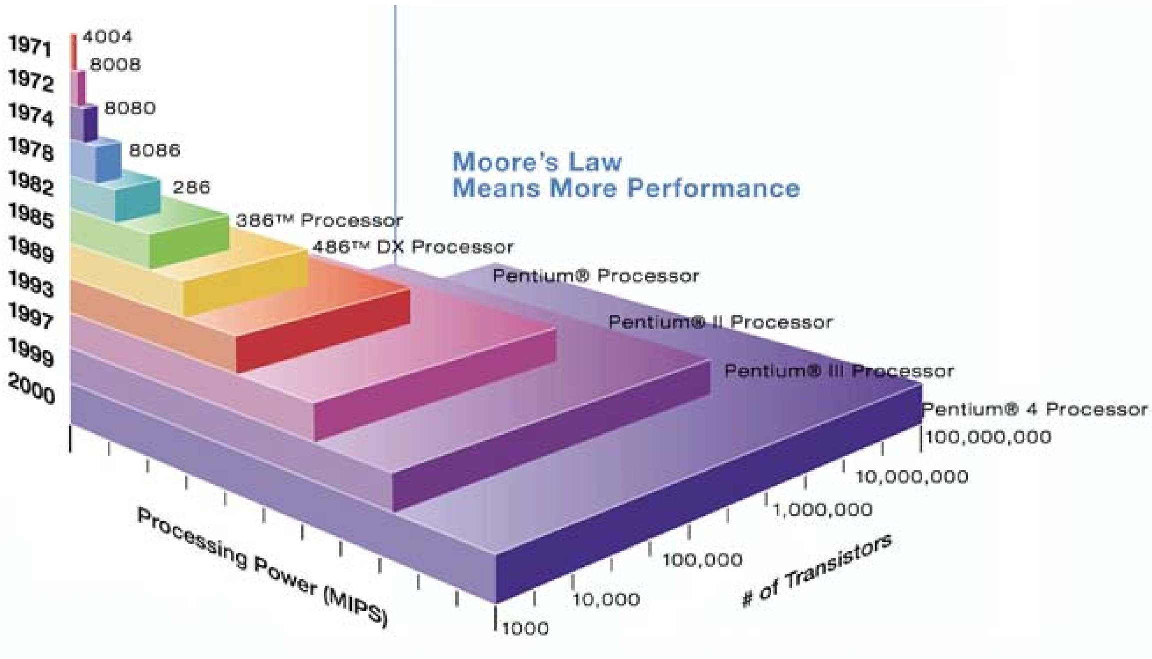

Figure 24. Moore's Law. (From [27])

After examining the Figure 24, it should become naturally evident why the military would look to smart sensors for more resourceful and adaptable operations. The only data missing that might emphasize this would be to have adjusted prices for each processor, although, it can be surmised that the prices either remain relatively the same if not dropping in price. Figure 24 shows how sensors (transistor axis could be thought of as the sensor axis as well), in parallel with microprocessors, are increasing in numbers and types. Since there will never be enough humans, intelligent centers, bandwidth and power (when mobile), these multitudes of sensors must be automated.

One of the most basic capabilities of the smart sensor is the ability to communicate in a non-analog fashion, so it is apparent that the communication interface of a sensor network is of utmost importance. Some of the motivations behind interfacing sensors and networks are: cost saving, remote monitoring, modularity, flexibility, and more accurate measurements at higher data rates. IEEE defines a smart sensor as a sensor "that provides functions beyond those necessary for generating a correct representation of a sensed or controlled quantity. This function typically simplifies the integration of the transducer into applications in a networked environment [28]." The

key identifier of a smart sensor is the fusing of the sensor with the information processing and communication technologies, giving it more capabilities than the standard raw data sensor [26].

1. Basic Sensor Architecture

A simple architecture for a smart sensor was proposed by Robert Johnson and Stan Woods to be included with the IEEE 1451.2 Standard for Smart Transducers (Figure 25) [52].

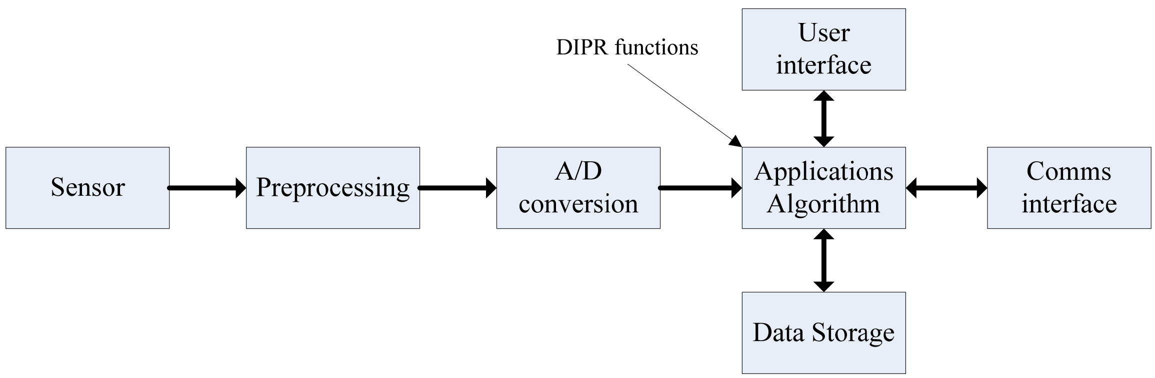

Figure 25. Basic Architecture of a Smart Sensor. (After [52])

a. Transducer

The part of the device that converts energy from one domain into another (may be a sensor or an actuator).

b. Signal Pre-processing

This includes the circuitry that prepares the electrical signal for conversion to the digital domain (some sensors do not require this step).

c. Analog-to-digital (A/D) Conversion

As implied, this is the step in which a device converts the raw sensor data into a digital code that represents the magnitude of the analog signal. It is considered beneficial to perform this conversion as close to the point of measurement as possible.

d. Application algorithms.

This is the "brains" where the sensor gets its "smartness." It is the application layer of software or hardware with functions that include converting the digital data to units specified by the user, signal processing, data analysis and reduction, monitoring alarm conditions, time stamping data strings, or other operations that are deemed necessary to be performed closely to the point of measurement. An example of an application algorithm is the detection function from the DIPR architecture mentioned in Chapter III.

e. User Interface

This is a standardized presentation of the data to the end user in the application specific units (GUI, alarm, etc). If the user interface is connected to a network sensor, this is a useful place where the results of the prediction function is presented and decision maker can react accordingly (rules of engagement).

f. Data Storage

Sensor data that is essential to a successful sensor network includes: sensor identification and configuration information; calibration data; history of data time stamps; and much more depending upon the size of the memory and amount of information that is able to be communicated through available bandwidth. Ideally, the storage is located on a network server.

g. Communication Node

An interface is required to allow for communication and remote access to the sensor for setup, calibration, diagnostics, status monitoring, and obviously data capture.

2. Systems Engineering Perspective

The systems engineer needs to be aware of the core capabilities and central functionalities available in a smart sensor [26].

a. Two-way Digital Communications

This capability cannot be over-emphasized. It should be considered the number one parameter. Other important functions are dependent upon the communication function be supported. An example of a secondary function is remote configuration and calibration. Additionally, smart sensors need to be scalable, re-configurable, and updated. Standard communication interfaces are crucial in assessing the communication capability of a smart sensor (to be discussed further later).

b. Self-identification

The smart sensor must know what it is and how it works (what the goal/mission of the sensor is). It is also important for the sensor to communicate this information with other components via the network. This is a key capability of plug-and-play sensors (sensors that can be integrated into a system with a standardized interface, such as USB-interfaced sensors).

c. Self-diagnosis

With the massive amount of available sensors, it is physically impossible for maintenance personnel to be able to set up any kind of plan to regularly check every sensor in a system, especially systems as complex and distributed as those that exist in the military. The solution is to be able to have smart sensors capable of notifying a maintenance person or user when it is no longer operating within parameters (or more ideally, about to fail). Such a self diagnosis would ideally cover the sensing element, all electronics, power supply, wiring and the physical body or casing, and other components (depending upon the needs/requirements of the user).

d. Powerful Data Processing

As microprocessors become more inexpensive and smaller, they are becoming more prevalent and embedded on sensors of all types. The more “powerful” (in computing ability) a microprocessor is, the greater the ability to infuse the sensor with

smarter capabilities, more advanced algorithms and AI. The powerful data processing makes the above mentioned capabilities more efficient and relevant. Various sensor protocols will be discussed next.

3. Sensor Protocols

It is important for the systems engineer to understand that it is not necessarily paramount to have a standard sensor, but, it is of the utmost importance to have standard protocols and interfaces for sensors. Having a standard protocol is important for a variety of reasons: provides for modularity; easier installation of sensors; future expansion of the sensor network; and perhaps most importantly, simplicity in design. The network-centric systems engineer involved with smart sensor networks needs to be familiar with a variety of sensor protocols in order to make an accurate assessment of the most appropriate protocol required for a particular system. Systems engineers and the DoD must emphasize (and push) the importance of a standard protocol throughout industry. Below are various protocols and buses in use throughout the sensor industry.

a. Actuator Sensor Interface (ASI)

ASI was developed in Germany by a group of sensor manufacturers. It is a low-cost, bit-level system, designed to handle only four bits per message in a basic architecture with a minimal operating range (mainly designed for factory automation and process control environments) [53].

b. Highway Addressable Remote Transducer (HART)

HART is a network promoted by Rosemount Inc. (a manufacturer of high accuracy sensors). HART is a "bi-directional communication protocol that provides data access between intelligent field instruments and host systems. A host can be any software application, from a hand-held device or laptop to a process control or asset management [29]."

c. Foundation Field Bus

Foundation field bus (FF) was intended to be a replacement for the 4-20 mA standard (used for analog signaling), but it has experienced numerous delays during its development by the Instrumentation, Systems and Automation Society (ISA). FF is meant to be used for applications using basic and advanced control. FF is mainly utilized by the manufacturing industry [54].

d. Process Field Bus

Process field bus (Profibus) is another German developed standard for field bus communications. There are two different types of Profibus: Decentralized Peripherals (DP) and Process Automation. Profibus DP is the more widely used variation. It is used for the operation of sensors and actuators through a central controller. It can be used to network multiple controllers to one another [55].

e. Standardization

The above protocols (and many others) have emerged as accepted industry standards for connecting sensors, input and output devices, and other smart devices to a network. Ideally, smart sensor interfaces should conform to a worldwide standard. The obvious difficulty in creating such a worldwide standard is that many vendors are reluctant to support a single standard in fear of losing their competitive edge. This is why the DoD needs to be one of the driving forces behind standardization for sensor protocols for network-centric systems (not in the sense of pushing a certain standard, but, that some sort of standard be set).

The core goal of standardization should be interoperability in a wide range of operations. The IEEE 1451 committee has been spearheading an adoption of standards since 1997. One of their first standards approved, IEEE Std 1451.2-1997, is titled “IEEE Standard for a Smart Transducer Interface for Sensors and Actuators—Transducer to Microprocessor Communication Protocols and Transducer Electronic Data Sheet Formats”. Any network-centric systems engineer should have this reference when working on a project involving sensor networks [28].

The main objectives of the IEEE standard are to [28]:

- “Enable plug and play at the transducer (sensor or actuator) level by providing a common communication interface for transducers.”
- “Enable and simplify the creation of networked smart transducers.”
- “Facilitate the support of multiple networks.”

The above objectives are (or should be) the same objectives of the network-centric systems engineer when developing a smart sensor network. When the engineer utilizes the IEEE standard, they will be able to isolate the wide array of choices of transducers from the choice of networks. Additionally, standardization makes prototyping a wide array of sensors easier (more attention can be given to the actual desired of the function).

In this section, the essentials to understanding sensor networks were discussed; the next section provides information on data fusion.

C. DATA FUSION

Combining various data from sensors (both similar and dissimilar sensors) is sensor fusion and combining that data together with current-state and a priori knowledge can be considered information fusion [30]. Reference [31] defines sensor fusion as “the combination of information from different sensors to capture data of the environment whose obtaining is beyond the capacity of each sensor individually, mainly when reliability and precision are considered.” Sensor fusion basically makes any system more tolerant to faults and can make available new information that none of the sensors alone could supply. Figure 26 shows the basic architecture of sensor fusion.

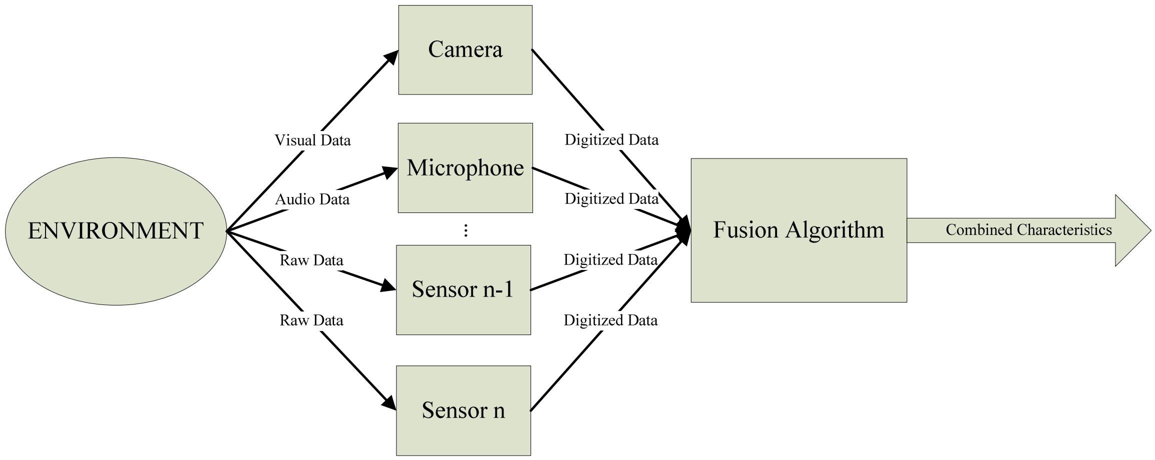

Figure 26. Basic Architecture of Sensor Fusion. (After [31])

Four different types of sensor fusion are specified below [31]:

1. **Complementary Fusion**

Different types of sensors provide different (complementary) views of the environment (e.g., video and motion data provide more accurate data on a target).

2. **Competitive Fusion**

The goal of competitive fusion is to provide redundant information about the area of the environment observed. More than one sensor observes the same feature of the environment (e.g., two cameras observing the same space and comparing figure of merit for accuracy of data).

3. **Cooperative Fusion**

Cooperative fusion is the combination of data from independent sensors in order to obtain information that could not be obtained by any of the sensors by themselves (e.g., a camera working in conjunction with a robot to obtain information on a target's identity).

4. **Independent Fusion**

Unrelated sensors provide information to a common storage location in a common data structure (e.g., the building of a knowledge base for future learning of behavior).

In this section, different types of data fusion were presented; the next section provides information on some advanced algorithms used for data fusion.

D. ADVANCED FUSION ALGORITHMS

There are numerous types of algorithms that have been used for the fusion process, such as, Bayesian Networks, Dempster-Shafer method and Artificial Neural Networks (as well as combinations of any number).

1. Bayesian Network

The Bayesian network algorithm forms the basis for many modern algorithms with numerous variations. The basis for Bayesian networks lies within the equation known as Bayes' rule:

$$P(b \mid a) = \frac{P(a \mid b)P(b)}{P(a)},$$

where P(a) and P(b) are the marginal probabilities of a and b, respectively, and P(a|b) and P(b|a) are the conditional probabilities. A simple example to help in understanding Bayes' rule is to consider the weather (this is applicable since it involves the real world/environment and sensors) where the goal is to determine the probability of rain, given there are clouds in the sky. Consider the following probabilities:

$P(\textit{clouds in sky}) = 0.50$ or $P(a)$,

$P(\textit{rain}) = 0.30$ or $P(b)$,

$P(\textit{clouds in sky} \mid \textit{rain}) = 0.90$ or $P(a|b)$.

Which results in:

$$P(\textit{rain} \mid \textit{clouds in sky}) = \frac{P(\textit{clouds in sky} \mid \textit{rain})P(\textit{rain})}{P(\textit{clouds in sky})} = \frac{(0.90)(0.30)}{0.50} = 0.54$$

Or, when there are clouds observed in the sky, there is a 54% chance it will rain.

Bayes' rule may not seem helpful but over time in a system that is updatable or can learn, a system that utilizes Bayes' rule becomes more reliable (given reliable data,

i.e., more data over time on the prior probabilities, P(clouds in sky) and P(rain)). A Bayesian network is used to represent the dependencies among variables and to give a concise specification of any full joint probability distribution. The specifics of the Bayesian network are not what is important (for the interested reader, they are directed to [8] for an in-depth description), what is important is to know what the Bayesian network can be used for. The Bayesian network is suited for an application in which prediction and/or learning is important.

2. Dempster-Shafer Method

The Dempster-Shafer theory was developed to deal with discerning the difference between "uncertainty and ignorance." Instead of calculating a certain probability associated with a problem, Dempster-Shafer determines the "belief function," Bel(X), which is the probability that the evidence supports the solution. Again, for the specifics of the Dempster-Shafer theory, the reader is directed to [8]. The perceived importance of the Dempster-Shafer theory is in its ability to aid in determining when more information (evidence) is required to support a conclusion (or solution).

3. Artificial Neural Networks

The motivation behind the creation of the artificial neural network was to create a mathematical model that mirrored the design of the human brain (composed of neurons). The study of neural networks is practically a whole other field of research in and of itself (refer to [8] for an in-depth description). The usefulness of an artificial neural network is that it can also be used as an inference algorithm and pattern recognition.

4. DIPR Advanced Fusion

Traditionally, sensor fusion takes place in the "Identification" subsystem of the DIPR standard "when intelligent rules combine (fuse) features into an intelligent state (symbol) [4]". In the case of advanced fusion for identification, feedback from the prediction or reaction subsystems might result in additional symbols, behavior labels, or inferred predicted outcomes. For example, an AI system that calls for action that it might normally based on the basic information may not make sense when the given information

is fused with lower level features. Take the following "real world" example into consideration: A sensor network embedded in the engineering space of a ship detects both excessive heat and smoke in the space, interprets this as a fire in the space, predicts that this will affect mission effectiveness, and reacts by releasing halon into the space (even though there are people still in the space and halon is harmful to humans). If the system had low-level data indicating that there were humans in the space still, it would "know" not to do this, and to wait until it sensed that all humans had evacuated the space.

In this chapter, the fundamentals of sensor networks were discussed; the next chapter will present applications of sensor networks and artificial intelligence.

V. APPLICATIONS

A. OVERVIEW

As shown, there are various ways of modeling systems with artificial intelligence. The methods presented by this thesis are not the only methods that have been researched and studied. Below are various applications of network-centric systems of systems that use AI and smart sensor networks (including academic papers) to enhance network-centric operations (proofs of concept in which AI and smart sensor networks enhance NCW).

B. APPLICATIONS

Brief overviews of the below applications are presented in this section. For more information on each one, refer to the applicable references.

1. NPS NCSE Watchman Project

This was a student project that followed the systems engineering process closely, from the definition of a need, identification of requirements, system design, system implementation and system testing. The project laid the groundwork for further research to be conducted on a network-centric system. Figure 27 shows the system architecture for the Watchman project. It shows the distributed sensors (WiLife cameras) and their connections through the network nodes to the central database. The Watchman project followed a systems engineering approach to creating a network centric system. Additionally, it embodied the principles and functions of DIPR [32]. Other systems have been and are in the process of being added to the NPS network-centric AI system of systems.

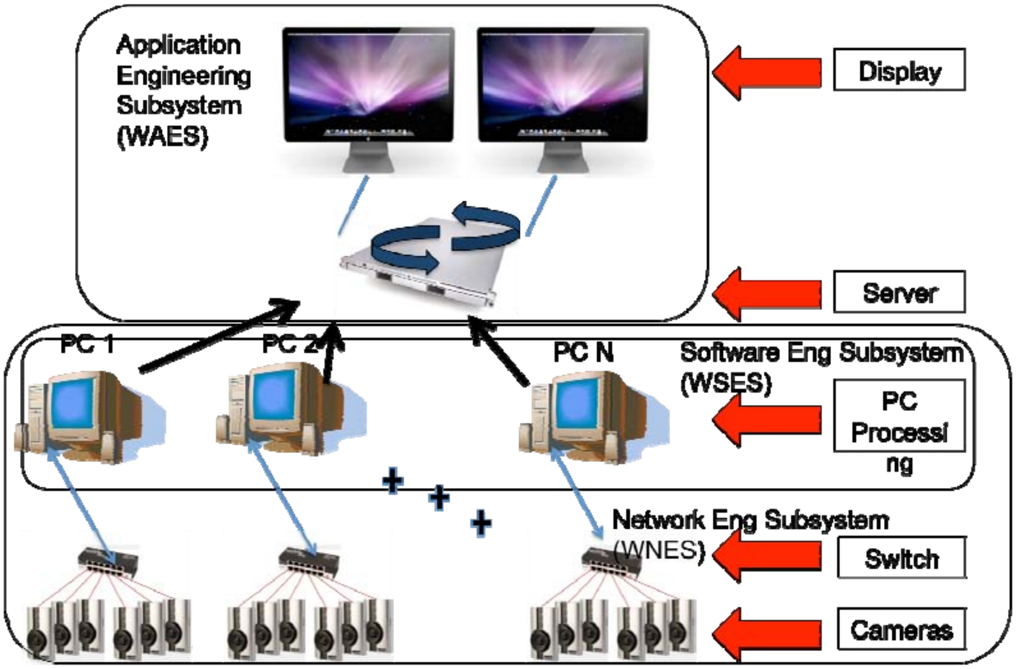

Figure 27. Watchman Architecture. (From [32])

2. Wireless Sensor Networks Lab

This Stanford University project, led by Professors Andrea Golsdmith and Hamid Aghajan, focuses on multi-camera networks, human interfaces, smart environments, user-centric design, and ambient intelligence. This laboratory studies the results of distributing sensors on a network. Figure 28 shows a schematic of the architecture for the user-centric environment discovery in a smart environment [33].

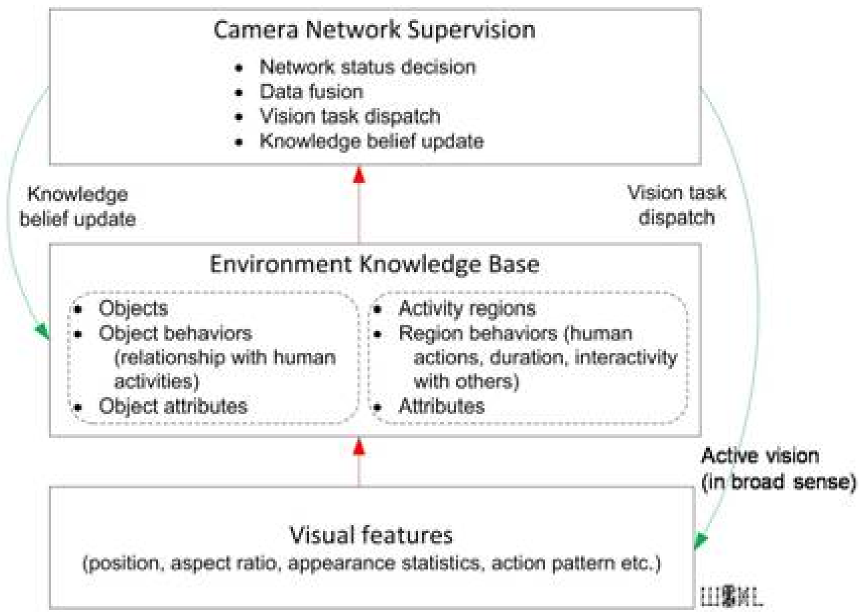

Figure 28. WSNL User-centric Architecture. (From [33])

3. Institute for Software Integrated Systems (ISIS)

This project is being conducted by engineers at Vanderbilt University. They are currently working to integrate a sensor network with soldiers' helmets to identify the location of possible snipers using acoustic information combined from sensors on the soldiers' helmets [34]. This project illustrates the value of sensor data obtained from numerous sources in providing a more accurate picture of the battle space. Figure 29 shows a screen shot of the prototype system. The green dots show the senor locations and the red dot shows the estimated location of the sniper [35].

Figure 29. ISIS Counter-sniper System. (From [35])

4. Ambient Intelligence

Ambient intelligence, a relatively new area of research, focuses on enabling and improving human interactions with a smart environment. The goal is to integrate ubiquitous computing in an intelligent manner where the human is the center of the environment. Ambient intelligence encompasses all aspects of AI and sensor networks. This area of research still has much to grow, but the possibilities associated with it are limitless. More detailed information on ambient intelligence can be found in [36], [38]–[40]. Figure 30 illustrates an application of ambient intelligence designed specifically for assisting the elderly or sick in their home environments.

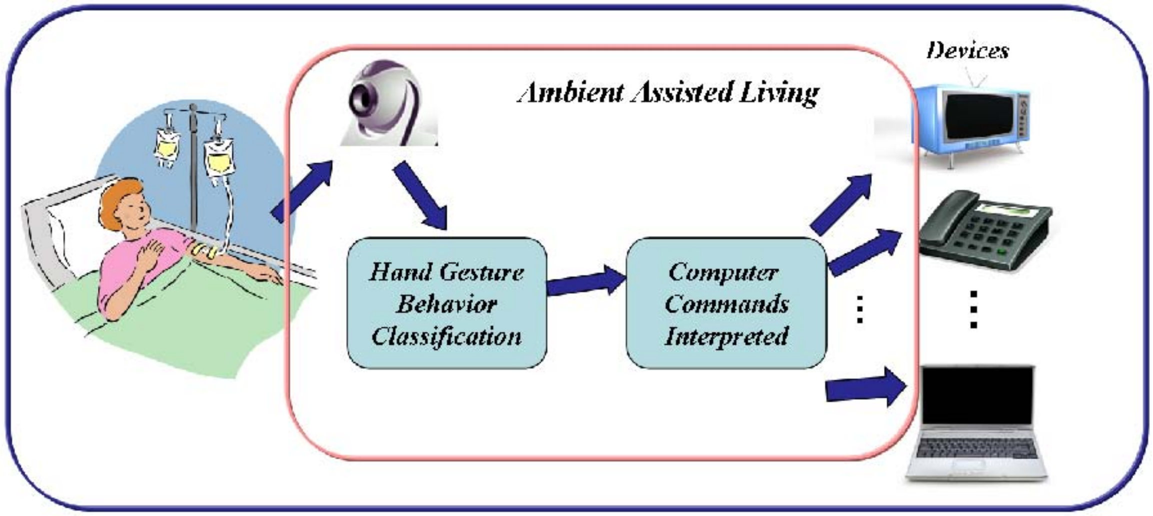

Figure 30. Ambient Assisted Living Application. (From [4])

5. High-tech Security

Wired magazine published an article titled "NYC is Getting New High-tech Defense Perimeter" in April 2008. The article discusses New York City's efforts to install 3,000 cameras into the financial district [41]. "The new cameras will be fully networked, with video-intelligence algorithms that aim to spot potential attackers before they perpetrate their crimes [41]." This example shows that large scale sensor networks of thousands of sensors are not only applicable to the military but also to the civilian sector (who have recognized the importance of sensor networks in enhancing security).

6. Highway Incident Management

The University of California, San Diego, has numerous projects that have been conducted to advance research in the field of AI and sensor networks. One such project, titled "Autonomous Agents for On-Scene Networked Incident Management (ATON)", has the goal to "make tangible and substantive contributions to the realization of a powerful and integrated traffic-incident detection, monitoring and recovery system [42]." This is an excellent example of a project in which sensors of different types were used. A schematic of the layout of sensors for this project is shown in Figure 31.

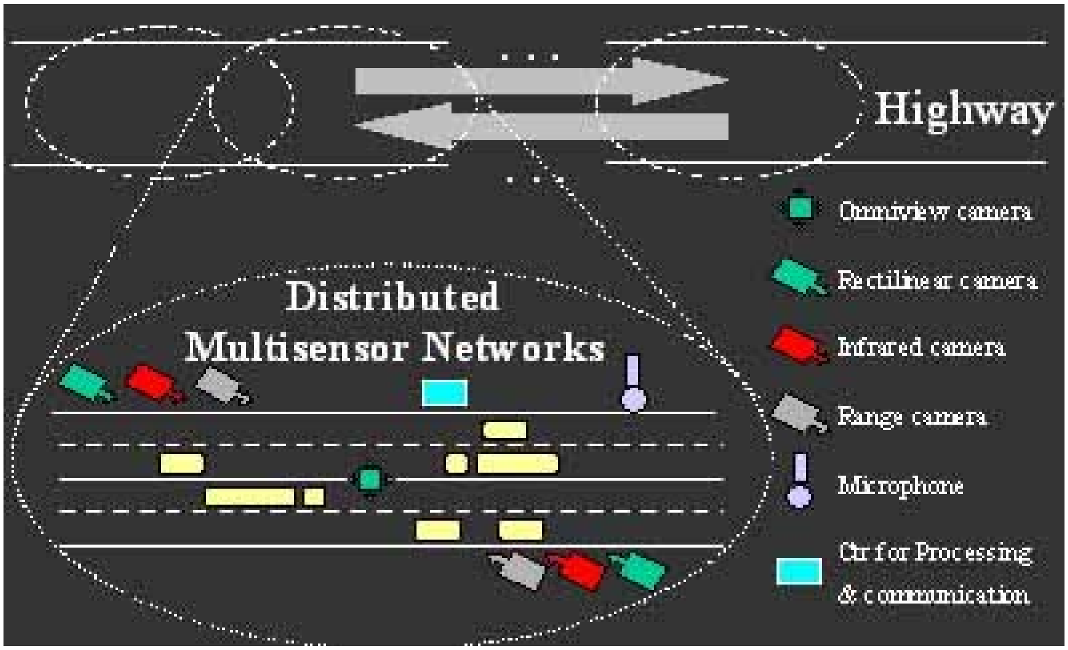

Figure 31. Diagram of Distributed Highway Video Sensors. (From [42])

7. "Distributed Bayesian Target Identification Algorithm"

This is an academic paper that exhibits that research in algorithms (even Bayesian based algorithms) is still evolving. The paper examines and compares two different algorithms: Hierarchical Bayesian target identification (HBTI) and Distributed Bayesian target identification (DBTI) [43]. This paper shows that advanced algorithms are of utmost importance in developing a smart sensor network. Figure 32 illustrates the architecture for DBTI.

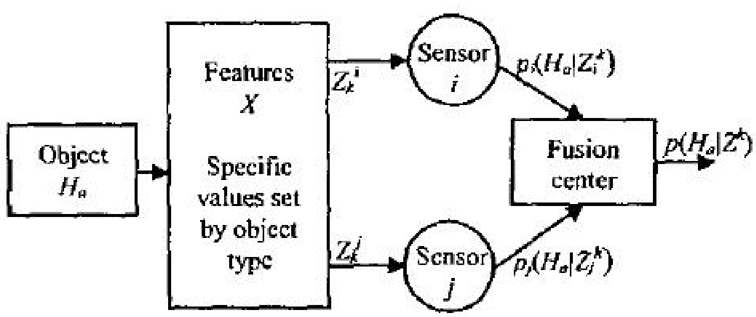

Figure 32. Distributed Bayesian Target Identification Multi-sensor System. (From [43])

8. "Real-Time Communication for Smart Sensor Networks: A CAN Based Solution"

This is an IEEE sponsored academic paper that examines the aspects of a controller-area network (CAN) [44]. The paper advocates that "CAN networks can be an interesting solution to support event-triggered smart sensor networks [44]." Three strategies for scheduling were developed for the management of the software-implemented outgoing communication queue, as shown in Figure 33. This paper illustrates the application of a sensor protocol in the design of a smart sensor network system.

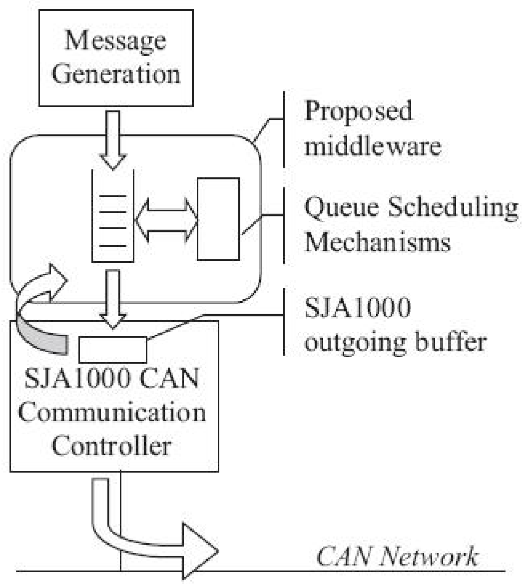

Figure 33. Remote Node with Scheduling Strategies. (From [44])

9. "Network Centric Multi-Agent Systems: A Novel Architecture"

This is an academic paper published by The Artificial Life and Adaptive Robotics Laboratory (ALAR) that proposes "a novel network centric multi-agent architecture (NCMAA), which is purely based on network theory, scales well, and provides a powerful engine [45]." This paper examines a network-centric MAS by focusing on modeling and simulation. The paper presents a two-layer architecture depiction of a network-centric MAS. Figure 34 shows the schematic of the NCMAA architecture.

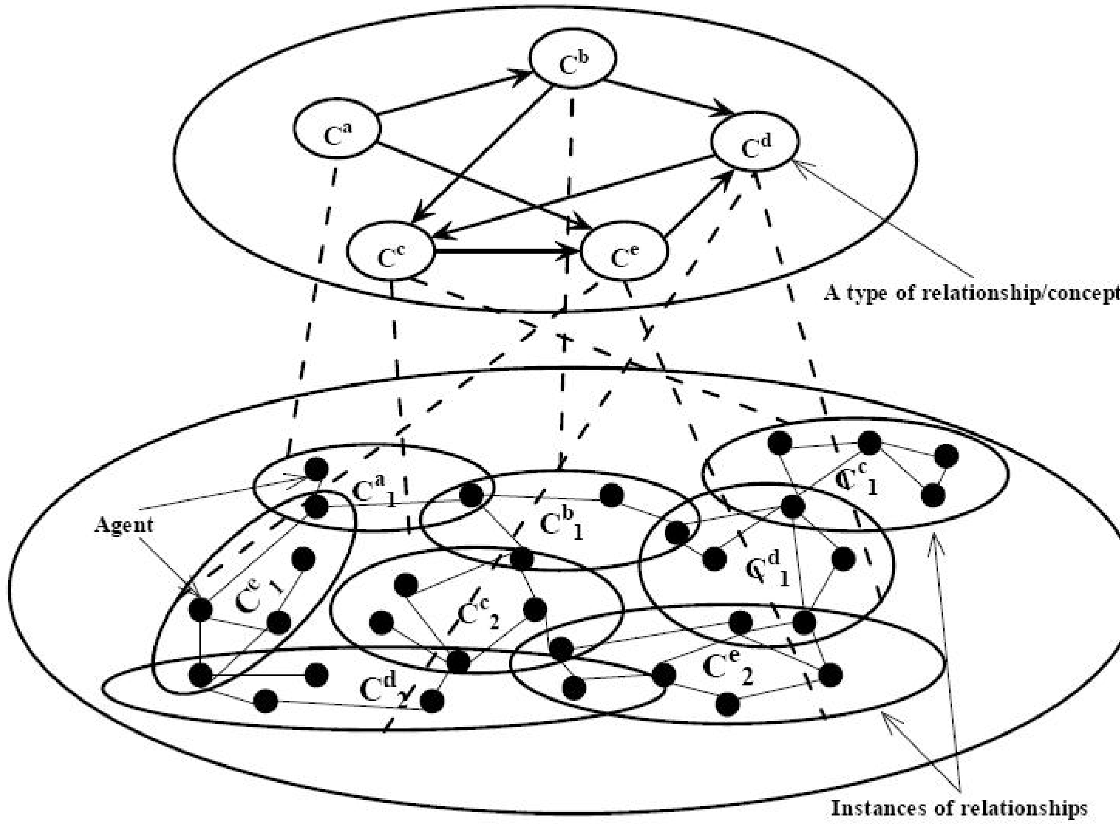

Figure 34. NCMAA Two Layer Architecture. (From [45])

In this chapter, snapshots of various applications of AI and sensor networks were presented; the next chapter will provide recommendations based upon the information presented by this thesis.

VI. RECOMMENDATIONS

Based on the information provided in this thesis, below are a set of recommendations for both system design of and education for smart sensor networks.

A. SYSTEM DESIGN

This section presents information that can be used in the design of a smart sensor network system. Operational concept scenarios are examined, an external systems diagram is presented, requirements are suggested, and a functional architecture is then offered.

1. Operational Concept

Reference [46] defines an operational concept as "a vision for what the system is (in general terms), a statement of mission requirements, and a description of how the system will be used [46]." Below are examples of these elements (hypothetical).

a. Vision

This smart sensor network system will be developed by Company X and produced by Company Y. The system is to be deployed in five years to be used by all ships in the U.S. Navy. It will be operated and maintained by crews of Navy ships. The system must be capable of being updated in both hardware and software to meet evolving operational needs and objectives. It will be operated for at least 20 years, and be phased out of service and disposed of by Company Y. The system will provide automated security coverage by detecting, identifying, predicting and reacting to input from the environment to accommodate manning reductions of the U.S. Navy.

b. Statement of mission requirements

"The mission requirements are stated in terms of measures of effectiveness (MOEs) [46]." Below are high-level examples of MOEs for the hypothetical case discussed in the previous sub-section:

- X% of the security performed by a ship's crew is able to be replaced by this system.
- Manning on board a ship was reduced by Y%.
- Manning across the Navy was reduced by Z%.
- The system is able to detect abnormal behavior T% of the time.
- Mission effectiveness increased by U%.

c. Operational Concept Scenarios

Below are some operational concept scenarios that might be associated with a smart sensor network:

- A user (onboard a ship) wants to add a sensor to an already existing network.
- A user wants to add a new AI algorithm for feature detection and interpretation.
- A commander needs to distribute to all nodes an updated knowledge base and an updated as well as an algorithm that will notify the commander to react to various sensor output from all nodes.
- Notification (by sensor) to maintenance personnel to perform repairs, replace power supply, or update firmware, etc.
- Initiate clock synchronization throughout the system (so all sensors have the same time stamps for data).
- Physical attributes of the world are obtained from the environment and processed in a way that is understandable to the user.

2. External Systems Diagram

The external systems' diagram is defined in [46] as "the model of the interaction of the system with other (external) systems in the relevant contexts, thus providing a definition of the system's boundary in terms of the system's inputs and output [46]." The external systems diagram, as illustrated in Figure 35, is based on the links between the elements of NCSE, and bounds the system design for smart sensor networks (top-level function "provide 'smart' information").

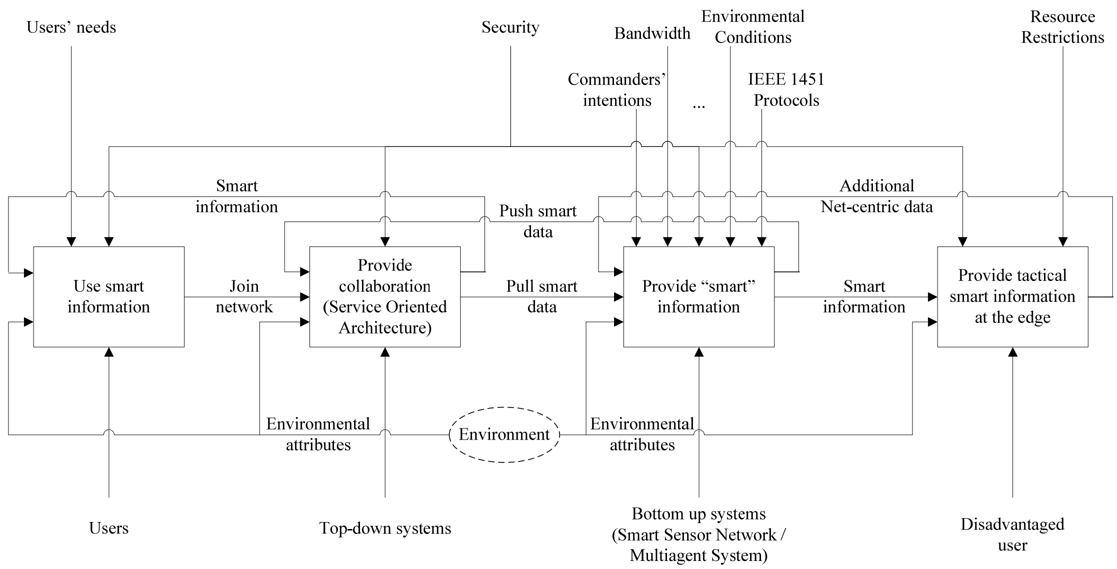

Figure 35. Smart Sensor Network System External Systems Diagram

The external systems for smart sensor networks are the user, collaborative technology systems, the disadvantaged user, and inherently, the environment. Some sample constraints for a smart sensor network (arrows going in from the top, in the above diagram) are security limitations, bandwidth availability, environmental conditions (e.g., weather), and IEEE 1451 protocols. The inputs are environmental attributes (raw data), requests by collaborative technologies, and pertinent data needed to help the disadvantaged user. The ultimate output of the smart sensor network is smart information and data.

3. Requirements

The requirements of the stakeholders of the network-centric system (end-user, commanders, contractor, etc.) need to be understood so that the needs and the objectives of all involved are accomplished. Below are the standard systems engineering requirements; high level requirements for the design of a smart sensor network are listed in further detail.

- Input/output requirements (conform to IEEE 1451 interface and protocols/standards, what information is required by the user and what format will the information be output in, etc.).
- Technology and system-wide requirements (more sensors will be added to the network in the future, remain within budget/schedule, suitability, etc.)
- Trade-off requirements (e.g., performance vs. cost tradeoffs)
- System qualification requirements (e.g., ensure that the above requirements are met)
- AI requirements (e.g., the requirements to complete the DIPR cycle are met)

Below are the recommended high-level requirements for the design and operation of a generalized smart sensor network system.

C0 – Requirements

C1.0—Input/output requirements

C1.1—Input requirements

C1.1.1—The smart sensor network system shall receive queries from the top-down systems of collaboration (e.g., SOA).

C1.1.2—The smart sensor network system shall receive inputs from environmental attributes.

C1.1.3—The smart sensor network system shall receive additional network-centric details from disadvantaged users who have tactical smart information at the edge.

C1.1.4—The smart sensor network system shall receive raw data and environmental attributes from the environment.

C1.2—Output requirements

C1.2.1—The smart sensor network system shall provide smart information as an output.

C1.2.2—The smart sensor network system shall provide smart information as an output to the disadvantaged users at the tactical edge.

C1.2.3—The smart sensor network system shall provide smart information as an output to the top-down systems of collaboration (SOA).

C2.0—External systems requirements

C2.1—The smart sensor network system shall interface with end users.

C2.2—The smart sensor network system shall interface with collaborative technologies.

C2.3—The smart sensor network system shall interface with disadvantaged users.

C3.0—System constraint requirements

C3.1—The smart sensor network system shall comply with constraints of IEEE 1451 protocols.

C3.2—The smart sensor network system shall comply with the combatant commanders' intentions.

C3.3—The smart sensor network system shall comply with imposed security constraints.

C4.0—AI system requirements

C4.1—The smart sensor network system shall detect raw data from the environment and output spatial-temporal features.

C4.2—The smart sensor network system shall identify features and spatial-temporal attributes and fuse them into intelligent states.

C4.3—The smart sensor network system shall predict behaviors and infer outcomes from the behaviors.

C4.4—The smart sensor network system shall react by affecting an action on the environment.

C5.0—Technology and system-wide requirements

C5.1—The smart sensor network system shall overlay onto an MAS infrastructure.

C5.2—The smart sensor network system, at minimum, shall have components that include sensors, network backbone, software and other hardware.

C5.3—The smart sensor network system shall predict behaviors and infer outcomes from the behaviors.

C5.4—The smart sensor network system shall react by affecting an action on the environment.

C6.0—Trade-off requirements

C6.1—The smart sensor network shall incorporate compression through DAI as part of the tradeoff design.

C6.2—The smart sensor network shall incorporate bandwidth, power, weight, facilities, human analysts needed as part of the tradeoff design.

C6.3—The smart sensor network shall consider incorporate as part of the tradeoff design.

C6.4—The smart sensor network shall incorporate real-time processing as part of the tradeoff design.

C6.5—The smart sensor network shall incorporate the physical network distribution and the DAI overlay as part of the tradeoff design.

C7.0—System qualification requirements

C7.1—The qualification system for smart sensor networks shall prove DIPR functionalities.

C7.2—The qualification system for smart sensor networks shall prove infrastructure.

C7.3—The qualification system for smart sensor networks shall prove compression.

C7.4—The qualification system for smart sensor networks shall prove value to the network-centric mission.

4. Functional Architecture

Functional architecture is defined in [46] as "a logical architecture that defines what the system must do, a decomposition of the system's top level function [46]." "The functional architecture of a system contains a hierarchical model of the functions performed by the system [and] the system's components [46]." The diagram shown in Figure 36 is a general functional architecture that can be followed when designing a smart sensor network. The functional diagram of the smart sensor network integrates the functions of DAI and sensor networks.

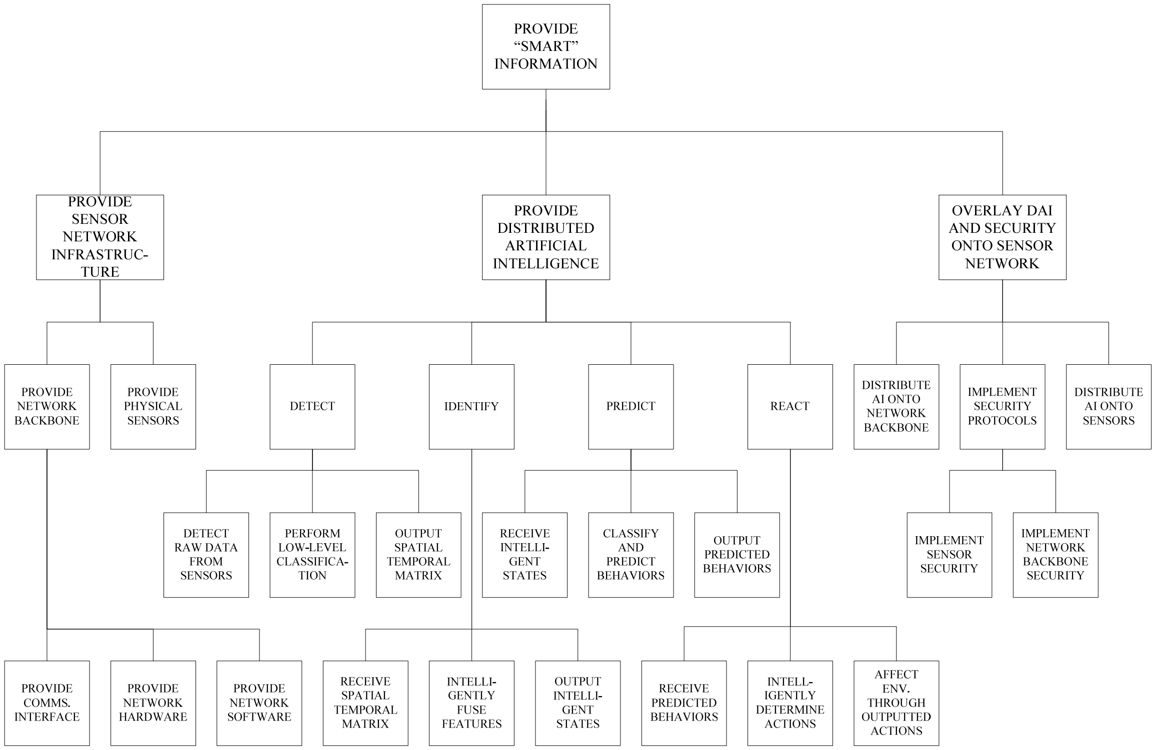

Figure 36. Smart Sensor Network Functional Architecture.

The architecture in Figure 36 begins with the primary function of providing smart information. The first subfunction identified is to provide sensor network infrastructure. Under this are the sub-subfunctions of providing physical sensors and providing the network backbone. Providing network backbone is further decomposed into providing a communications interface, network hardware, and network software. The next subfunction identified is to provide distributed artificial intelligence. Under this are four sub-subfunctions. The detect sub-subfunction is decomposed into: detect raw data from sensors; perform low-level classification; and output spatial-temporal matrix. The identify sub-subfunction is decomposed into: receive spatial-temporal matrix; intelligently fuse features; and output intelligent states. The predict sub-subfunction is decomposed into: receive intelligent states; classify and predict behaviors; and output predicted behaviors. The react sub-subfunction is decomposed into receive predicted behaviors; intelligently determine actions; and affect environment through outputted actions. The final subfunction identified is to overlay DAI and security onto sensor network. Under this are three sub-subfunctions: distribute AI onto network backbone;

distributed AI onto sensors; and implement security protocols (decomposed further into implement sensor and network backbone security).

The above functional architecture can be used when conceptualizing the design of any smart sensor network system, regardless of size. This architecture can also be tied back to the architecture of the previously mentioned Watchman project. The Watchman project mirrored the above functional architecture by breaking into teams or subsystems based on the identified functions of the system. The subsystems were then integrated together to create an operational system. The functional architecture of Figure 36 can be scaled to a smart sensor network system of any size, whether it is in the lab or in the field as a component of network-centric warfare.

In this section, a proposed method of system design of a smart sensor network system was proposed; the next section provides recommendations for education and research necessary for understanding smart sensor networks.

B. EDUCATION AND RESEARCH

Smart sensor networks have been identified as the key to the successful implementation of network-centric warfare and operations. Only through comprehensive education and cutting edge research will the DoD become the leaders in the world of smart sensor networks.

1. Education

The information provided in this thesis can be used in the development of a course based on the bottom-up approach, titled "Smart Sensor Networks for Network-Centric Systems Engineering." Below is a basic outline of this possible course.

a. Learning Objectives

Upon completion of a course in AI and smart sensor networks, the network network-centric systems engineer shall be able to:

- Design a smart sensor network system, given mission needs and objectives.

- Identify input/output requirements for a smart sensor network system.
- Recognize and identify the external systems with which the smart sensor network system interfaces.
- Identify and respond to constraints placed upon the smart sensor network system (industry standards, DoD standards, security, etc.).
- Understand the role AI plays in the design and development of a smart sensor network system.
- Understand the concepts of MAS and DAI, and use these concepts in the design of a smart sensor network system.
- Prove that the functions of DIPR in the creation of a smart sensor network system.

b. Schedule

The schedule of a network-centric systems engineer studying AI and sensor networks will have to be fast paced and will cover much material. (It is assumed that the student will already have an understanding of NCSE concepts, NCW, fundamentals of networking, probability and statistics.)

Week 1: History of AI/Introduction to AI (Ch. 1 of [8] and excerpts from [56])

Week 2: Introduction to agents and MAS/DAI (Ch. 2 of [8] and Prologue, Ch. 1, and Ch. 2 of [9])

Week 3: Search paradigms (uninformed, informed/ heuristics) and stochastic-based algorithms (Bayesian) (Ch. 3, Ch. 4, and Ch. 14 of [8] and excerpts from [49])

Weeks 4—5: Low-level classifiers (excerpts from [47])

Week 6: High-level classifiers and DIPR basics (excerpts from [47])

—Midterm exam (AI, classifiers and DIPR)

Week 7: Sensor basics and types (excerpts from [48])

—Begin outline of class project (create a smart sensor network, following systems engineering process, using MATLAB)

Week 8: Sensor networks (protocols and standards) (excerpts from [48] and [28])

Week 9: Design smart sensor network (use [47] as a starting point)

Week 10: Implement smart sensor network subsystems

Week 11: Integrate and test smart sensor network

Week 12: Presentations of results from project

—Final exam (Smart sensor networks)

c. Reading Material

In addition to this thesis and references, the below reading material are recommended for the student studying AI and sensor networks.

<u>Artificial Intelligence: A Modern Approach</u>, by Stuart Russell and Peter Norvig is the text widely accepted as the standard text for an introduction to AI [8].

<u>Computational Intelligence paradigms: Theory and Applications using MATLAB</u>, by S. Sumathi has not yet been published, but, it appears to provide comprehensive information on MATLAB [47].

<u>Knowledge Discovery from Sensor Data</u>, by Auroop R. Ganguly, et al, provides real world examples of sensor networks [48].

Probabilistic Reasoning in Multiagent Systems: A Graphical Models Approach, by Yang Xiang provides an in-depth and technical examination of Bayesian networks [49].

The Quest for Artificial Intelligence: A History of Ideas and Achievements, by Nils J. Nilsson provides a comprehensive study of AI by tracing the history of AI [56].

Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence, edited by Gerhard Weiss provides in-depth analysis of MAS and DAI [9].

"IEEE standard for a smart transducer interface for sensors and actuators - transducer to microprocessor communication protocols and Transducer Electronic Data Sheet (TEDS) formats," produced by IEEE provides information on standard definitions and protocols recommended for use [28].

2. Student Research

There is a multitude of student research possibilities associated with AI and sensor networks. Below are a couple possible research projects that could be performed by NCSE students.

a. Continuation of the "Watchman" Project

Watchman was a project initiated by students in the NCSE path in early 2009, with the purpose of detecting human movements, analyzing the behavior pattern and determining if it was normal, abnormal or unknown (Identification and Prediction), and notifying the user through a graphical user interface of this analysis. Watchman also provided a facial recognition function for "automatic mustering" of students [57]. The continuation of the Watchman project in the areas of human behavior analysis would provide tremendous insights into pattern recognition functions. Another path the Watchman project could venture is to attempt to add more sensors to the entire network

(such as audio or motion detection sensors). Additionally, work done to integrate robots into the sensor network would prove very beneficial in the study of multiagent systems.

b. Campus-wide Smart Sensor Network Project

This project might have the potential of being a possible project that could be run by a PhD candidate. The ultimate goal is to use the systems engineering approach to create a smart sensor network throughout the campus of Naval Postgraduate School (NPS). The vision is to find a hypothetical "customer" on campus (possibly the Meteorology and Oceanography department) that requires the collection of some type of data/information (through sensors). This project could integrate research across numerous departments, campus-wide, with the Systems Engineering students taking the lead. The team would determine the requirement sand desired capabilities as defined by the customer.

The systems engineering process would need to be adhered to and the deployment of sensor motes, small deployable smart sensor platforms (see Figure 37 for an example of a sensor mote), across the campus for data collection would then follow. Possible data that could be collected might be associated with measurable environmental parameters (e.g., temperature, pressure, humidity), with a goal of predicting changes in a meteorological phenomenon (e.g., temperature drop of 10 degrees and a pressure drop means wind speed will increase). This can then be compared against the actual results and the effectiveness of the algorithms can be determined. The benefits of this type of research are numerous. It would give systems engineers the experience with working with a customer to provide a required network-centric system, as well as providing research opportunities for other fields, such as the study of microclimates.

Figure 37. Crossbow MICA2 Sensor Mote. (From [50])

c. Conferences and Symposia

Students are encouraged to attend various conferences and symposia to enhance what they have learned, as well as a way of introducing them to cutting edge research. Additionally, students are recommended to submit academic papers and present their research findings at these conferences, such as ones sponsored by the Association for the Advancement of Artificial Intelligence (AAAI), IEEE and the Journal of Ambient Intelligence and Smart Environments published by IOS Press.

In this chapter, recommendations for design of a smart sensor network system and education were presented; the next chapter wraps it all up in a final conclusion and summary of the thesis.

VII. CONCLUSION AND SUMMARY

A. CONCLUSION

The study of AI and sensor networks may seem somewhat overwhelming, but, when they are looked at through the lens of systems engineering they become much less so. AI and sensor networks are the tools necessary to fulfill the goals of NCW. NCW sets the requirements and smart sensor networks are the means to the end of information fusion of the forces.

The network-centric systems engineer will have a better understanding of how to tackle a design problem involving smart sensor networks armed with basic knowledge of AI and sensor networks. The network-centric systems engineer must be an expert in network-centric systems; however, they do not need to be an expert in the fields of AI and sensor networks, exclusively. The network-centric systems engineer must be able to intelligently interact with experts in AI and sensor networks; having a general understanding and knowing where to find the answers will facilitate this. The network-centric systems engineer that is able to do this will be better equipped in determining how to design a smart sensor network system.

B. SUMMARY

Artificial intelligence and smart sensor networks are critical to the future of NCW and NCO, especially with Global War on Terror applications. To mitigate GWOT threats we must automate detection, identification, prediction and reaction to globally distributed potential terror threats. This must be carried out through the smart sensor networks utilizing DIPR and MAS. Overall, this thesis has presented the basics of artificial intelligence and sensor networks as well as the connections both of these research fields have to NCW. This thesis also recommended a systems engineering design of artificial intelligence and smart sensor networks, and the way ahead for education of AI and smart sensor networks relative network-centric systems.

These recommendations were shown in this thesis in the previous six chapters. Chapter I provided the reader with details on network-centric systems and the various

approaches to NCSE, as well as background on NCW and NCO. Chapter II introduced the study of AI by discussing the history and background, basic concepts and algorithms, and an introduction to low- and high-level classifiers. Chapter III discussed the importance of distributed artificial intelligence and the applicability of multiagent systems. Chapter IV centered on the basics of sensor networks and data fusion. Chapter V provided brief snapshots of various AI and sensor network applications. Chapter VI showed how systems engineering fits into the design application of smart sensor networks, and made recommendations with respect to education and smart sensor networks. Finally, Chapter VII presented the ultimate conclusion and summarized the thesis.

LIST OF REFERENCES

[1] R. E. Goshorn, "Findings for network-centric systems engineering education," presented at the Military Communications (MILCOM) Conference, San Diego, November 2008.

[2] United States Dept. of Defense. Office of Force Transformation, *The Implementation of Network-Centric Warfare*, U.S. Government Printing Office, Washington, D.C., 2005.

[3] M. Fredriksson and R. Gustavsson, "Quality of service in network-centric warfare and challenges in open computational systems engineering," *Proceedings of the Twelfth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003. (WET ICE 2003)*, pp. 359–364, IEEE Press, New York, Pub. No. 1080-1383, June 2003.

[4] R. E. Goshorn, D. E. Goshorn, J. L. Goshorn, and L. A. Goshorn, "Behavior modeling for detection, identification, prediction, and reaction (DIPR) in AI systems solutions," in H. Nakashima, H. Aghajan and J. C. Augusto (Editors), *Handbook of Ambient Intelligence and Smart Environments*, vol. 1, Springer, in press, 2009.

[5] G. Winstanley, "Artificial intelligence support for systems engineering," *IEE Colloquium on IT Support for Systems Engineers*, pp. 6/1–6/5, IEEE Press, Ne York, Pub. No. 3844604, 1990.

[6] M. N. O. Sadiku, "Artificial intelligence," *IEEE Potentials*, vol. 8, 1989, pp. 35–39.

[7] T. M. Jones, *Artificial Intelligence: A Systems Approach*, Infinity Science Press, Hingham, MA, 2008.

[8] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed., Prentice Hall, Upper Saddle River, NJ, 2002.

[9] G. Weiss, *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, The MIT Press, Cambridge, MA, 1999.

[10] E. R. Davies, *Machine Vision: Theory, Algorithms, Practicalities*, 3rd ed., Morgan Kaufmann, San Francisco, 2005.

[11] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 3rd ed., Prentice Hall, Upper Saddle River, NJ, 2007.

[12] L. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition*, Prentice Hall, Upper Saddle River, NJ, 1993.

[13] L. Rabiner and B. H. Juang, "An introduction to hidden Markov models," *IEEE ASSP Magazine*, vol. 3, 1986, pp. 1-4.

[14] J. G. Proakis and D. K. Manolakis, *Digital Signal Processing: Principles, Algorithms, and Applications*, 4th ed., Prentice Hall, Upper Saddle River, NJ, 2006.

[15] C. M. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, New York, 1996.

[16] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, 4th ed., Academic Press, Burlington, MA, 2009.

[17] M. Sipser, *Introduction to the Theory of Computation*, PWS Pub. Co., Boston, 1996.

[18] D. E. Goshorn, "The Enhancement of Low-Level Classifications in Sequential Syntactic High-Level Classifiers," Computer Science PhD Research Exam, UCSD, August 2008.

[19] J. Boardman and B. Sauser, "System of systems—the meaning of of," *2006 IEEE/SMC International Conference on System of Systems Engineering*, pp. 118–123, IEEE Press, New York, Pub. No. 1-4244-0188-7, 2006.

[20] J. Ferber, *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*, Addison-Wesley Professional, London, 1999.

[21] B. Chaib-draa, "Industrial applications of distributed AI," *ACM Communications*, vol. 38 (1995), pp. 49–53.

[22] N. P. Padhy, *Artificial Intelligence and Intelligent Systems*, Oxford University Press, New Dehli, India, 2005.

[23] S. Ossowski, "Distributed Artificial Intelligence," in J.G. Carbonell and J. Siekmann (Editors), *Co-ordination in Artificial Agent Societies,* Springer, Berlin, 1999, pp. 31–63.

[24] M. Schumacher, "Multi-agent systems," in in J.G. Carbonell and J. Siekmann (Editors), *Objective Coordination in Multi-Agent System Engineering,* Springer, Berlin, 2001, pp. 9–32.

[25] N. Vlassis, "A concise introduction to multiagent systems and distributed artificial intelligence," in R.J. Brachman and T.G. Dietterich (Editors), *Synthesis Lectures on Artificial Intelligence and Machine Learning,* vol. 1, Morgan and Claypool, San Rafael, CA, 2007, pp. 1–71.

[26] Yong Zhang, Yikang Gu, V. Vlatkovic and Xiaojuan Wang, "Progress of smart sensor and smart sensor networks," *Fifth World Congress on Intelligent Control and Automation, 2004 (WCICA 2004)*, vol. 4, IEEE Press, New York, Pub. No. 0-7803-8273-0, pp. 3600–3606, 2004.

[27] P. DelVecchio, "De-mystifying software performance optimization," available: http://software.intel.com/en-us/articles/de-mystifying-software-performance-optimization/ (accessed July 8, 2009).

[28] Anonymous, "IEEE standard for a smart transducer interface for sensors and actuators - transducer to microprocessor communication protocols and Transducer Electronic Data Sheet (TEDS) formats," *IEEE Std 1451. 2-1997*, IEEE Press, New York, Pub. No. 1-55937-963-4, 1998.

[29] HART Communication Foundation, "What is HART?," available: http://www.hartcomm.org/protocol/about/aboutprotocol_what.html (accessed August 12, 2009).

[30] R. P. Bonasso, "What AI can do for battle management," *AI Magazine,* vol. 9, 1988, pp. 77–83.

[31] K. Faceli, A. C. P. L. F. de Carvalho, and S. O. Rezende, "Combining intelligent techniques for sensor fusion," *Proceedings of the 9th International Conference on Neural Information Processing, 2002 (ICONIP '02),* vol. 4, pp. 1998–2002, IEEE Press, New York, Pub. No. 981-04-7524-1, 2002.

[32] R. E. Goshorn, D. E. Goshorn, J. L. Goshorn, and L. A. Goshorn, "Abnormal behavior classification and alerting through detection, identification, prediction and reaction (DIPR) system applied to a multi-camera network," submitted to the Workshop on Behavior Monitoring and Interpretation: Moving Objects in a Three Dimensional World, November 2009.

[33] A. Goldsmith and H. Aghajan, "Wireless sensor networks laboratory," available: http://wsnl.stanford.edu/smartenv.html (accessed September 3, 2009).

[34] D. F. Salisbury. "Network turns soldiers' helmets into sniper location system," available: http://www.physorg.com/news157123967.html (accessed September 3, 2009).

[35] A. Ledeczi, "Countersniper system," available: http://www.isis.vanderbilt.edu/projects/countersniper (accessed September 3, 2009).

[36] J. C. Augusto and D. Shapiro, "Advances in ambient intelligence," *Frontiers in Artificial Intelligence and Applications*, vol. 164, 2007, pp. 1–186.

[37] H. G. Ramos, "A contribution to the IEEE STD. 1451.2-1997 revision and update," *IEEE Africa Conference* (*AFRICON 2007)*, pp. 1–7, IEEE Press, New York, Pub. No. 978-1-4244-0986-0, 2007.

[38] P. Remagnino, H. Hagras, N. Monekosso, and S. Velastin, "Ambient Intelligence," *Ambient Intelligence: A Gentle Introduction*, New York: Springer pp. 1–14, 2005.

[39] P. Remagnino and G. L. Foresti, "Ambient intelligence: A new multidisciplinary paradigm," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 35, 2005 pp. 1–6.

[40] C. Ramos, "Ambient intelligence – A state of the art from artificial intelligence perspective," in J.G. Carbonell and J. Siekmann (Editors), *Progress in Artificial Intelligence*, Springer, Berlin, 2007, pp. 285–295.

[41] N. Shachtman, "NYC is getting a new high-tech defense perimeter. Let's hope it works," available: http://www.wired.com/politics/security/magazine/16-05/ff_manhattansecurity (accessed September 3, 2009).

[42] University of California, San Diego, "Autonomous agents for on-scene networked incident management," available: http://cvrr.ucsd.edu/aton/ (accessed September 3, 2009).

[43] Mei Wei, Shan Gan-Lin, and Wang Hong-feng, "Distributed bayesian target identification algorithm," *Proceedings of the Sixth International Conference of Information Fusion, 2003*, vol. 2, pp. 1379–1383, IEEE Press, New York, Pub. No. 0-9721844-4-9, 2003.

[44] A. J. Pires, E. M. Leao, J. P. Sousa, L. A. Guedes, and F. Vasques, "Real-Time Communication for Smart Sensor Networks: A CAN Based Solution," *Industrial Informatics, 2007 5th IEEE International Conference on*, vol. 1, pp. 553–558, 2007.

[45] A. Yang, H. A. Abbass, R. Sarker, and M. Barlow, "Network centric multi-agent systems: A novel architecture," *The Artificial Life and Adaptive Robotics Laboratory (ALAR) Technical Report Series*, pp. 1–13, ALAR, Canberra, Australia, Pub. No. TR-ALAR-200504004, 2008.

[46] D. M. Buede, *The Engineering Design of Systems: Models and Methods*, 2nd ed., John Wiley & Sons, Inc., Hoboken, NJ, 2009.

[47] S. Sumathi and S. Paneerselvam, *Computational Intelligence Paradigms: Theory & Applications using MATLAB*, CRC Press, Boca Raton, FL, 2009.

[48] A. R. Ganguly, J. Gama, O. A. Omitaomu, M. M. Gaber, and R. R. Vatsavai, *Knowledge Discovery from Sensor Data*, CRC Press, Boca Raton, FL, 2008.

[49] Y. Xiang, *Probabilistic Reasoning in Multiagent Systems: A Graphical Models Approach*, Cambridge University Press, Cambridge, UK, 2002.

[50] Crossbow Technology, "MICA2 wireless module," available: http://www.xbow.com/Products/productdetails.aspx?sid=174 (accessed August 4, 2009).

[51] J. Markoff, “The coming superbrain,” available: http://www.nytimes.com/2009/05/24/weekinreview/24markoff.html (accessed July 11, 2009).

[52] R.N. Johnson and S. P. Woods, “Proposed enhancements to the IEEE 1451.2 standard for smart transducers,” available: http://archives.sensorsmag.com/articles/0901/74/main.shtml (accessed September 22, 2009).

[53] AS-Interface, “ASI homepage,” available: http://www.as-interface.net (accessed September 22, 2009).

[54] Fieldbus Foundation, “About Fieldbus Foundation,” available: http://www.fieldbus.org (accessed September 22, 2009).

[55] Profibus-Profinet, “Profibus homepage,” available: http://www.profibus.com/technology/profibus (accessed September 22, 2009).

[56] N.J. Nilsson, *The Quest for Artificial Intelligence: A History of Ideas and Achievements*, Cambridge University Press, New York, 2009.

[57] System-Level Design, “Artificial intelligence: this time it’s for real,” available: http://chipdesignmag.com/sld/blog/2008/12/29/artificial-intelligence-this-time-its-for-real/ (accessed September 22, 2009).

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Dr. Rachel Goshorn
   Naval Postgraduate School
   Monterey, California

4. Ms. Deborah Goshorn
   Naval Postgraduate School
   Monterey, California

5. Dr. Eugene Paulo
   Naval Postgraduate School
   Monterey, California